
HOMO RES SACRA

ՀԱՅ-ՌՈՒՍԱԿԱՆ ՀԱՄԱԼՍԱՐԱՆ

ԼՐԱԲԵՐ

ՀԱՅ-ՌՈՒՍԱԿԱՆ ՀԱՄԱԼՍԱՐԱՆԻ

ՍԵՐԻԱ

ՖԻԶԻԿԱՄԱԹԵՄԱՏԻԿԱԿԱՆ
ԵՎ ԲՆԱԿԱՆ ԳԻՏՈՒԹՅՈՒՆՆԵՐ

№ 1

ՀՌՀ Հրատարակչություն

Երևան 2021

РОССИЙСКО-АРМЯНСКИЙ УНИВЕРСИТЕТ

В Е С Т Н И К

РОССИЙСКО-АРМЯНСКОГО УНИВЕРСИТЕТА

СЕРИЯ:

ФИЗИКО-МАТЕМАТИЧЕСКИЕ И ЕСТЕСТВЕННЫЕ НАУКИ

№ 1

Издательство РАУ

Ереван 2021

Печатается по решению Ученого совета РАУ

Вестник РАУ, № 1. – Ер.: Изд-во РАУ, 2021. – 180 с.

Журнал входит в перечень периодических изданий,
зарегистрированных ВАК РА и РИНЦ

Российско-Армянский университет, 2021 г.

ISBN 1829-0450

МАТЕМАТИКА И ИНФОРМАТИКА

УДК 621.372.54.083.92

Поступила:.24.03.2021г.
Сдана на рецензию: 26.03.2021г.
Подписана к печати: 31.03.2021г.

ОБ ОДНОЙ ГИБРИДНОЙ СХЕМЕ НЕСМЕЩЁННОГО ФИЛЬТРА С КОНЕЧНОЙ ИМПУЛЬСНОЙ ХАРАКТЕРИСТИКОЙ И ФИЛЬТРА КАЛЬМАНА

А.А. Дарбинян, А.Р. Акопян, М.А. Хачатурян

Российско-Армянский Университет

alexander.hakobyan@gmail.com, arman.darbinyan@rau.am

АННОТАЦИЯ

В данной работе речь идет о том, что в задачах радиолокационной оптимизации часто возникает вопрос о выборе между фильтром с конечной импульсной характеристикой и фильтром Кальмана. Оба этих фильтра имеют свои сильные и слабые стороны. В данной работе для более оптимальной оценки позиции объекта рассматривается гибрид вышеуказанных двух фильтров.

Ключевые слова: фильтр Кальмана, Импульсный фильтр, Гибрид фильтров.

1. Введение и Обозначения

$\mathbb{R}^n$ – n-мерное вещественное векторное пространство;

A^T – транспонированная матрицы A;

I – единичная матрица;

$M[X]$ – мат. ожидание случайной величины X;

$Cov[X] := M[XX^T] - M[X]M[X^T]$ – ковариационная матрица случайного вектора X;

$tr(A) \coloneqq \sum_{i=1}^{n} a_{ii}$ – след квадратной матрицы $A = (a_{ij})_{i,j=1}^{n}$.

В работе рассматривается задача изменения состояния n-мерного объекта по полученным r-мерным данным от приемника. Обозначив через X_k его состояние в момент времени T_k, предположим, что оно изменяется по заданному закону:

$$X_{k+1} = F_k X_k + G_k W_k, \tag{1.1}$$

где F_k– $n \times n$ матрица, называемая «матрицей перехода», G_k–$n \times m$ – вещественная матрица, а $W_k m$ – мерный Гауссовский случайный вектор с нулевым мат. ожиданием. Обозначим $Q_k := Cov[W_k]$.

Допустим, что некоторое устройство измеряет данный объект по заданной модели:

$$\breve{X}_k = H_k X_k + V_k, \tag{1.2}$$

где $\breve{X}_k \in \mathbb{R}^r$ измерение k – того состояния, $H_k r \times n$ – мерная матрица, а$V_k r$ – мерный случайный Гауссовский вектор с нулевым мат. ожиданием. Обозначим $R_k \coloneqq Cov[V_k]$.

2. Фильтр Кальмана

Рассмотрим n-мерный вектор на основе полученных измерений.

$$\hat{X}_k = \tilde{X}_k + K_k(\breve{X}_k - H_k \tilde{X}_k), \tag{2.1}$$

где $\tilde{X}_k = F_{k-1}\hat{X}_{k-1}$ и K_k – $m \times n$ – вещественная матрица. Наша цель минимизировать $tr\big(Cov\big[X_k - \hat{X}_k\big]\big)$.

Теорема 1. (см: [1] или [2]). Если определить матрицу K_k по данной формуле:

$$K_k = \tilde{P}_k H_k^T (H_k \tilde{P}_k H_k^T + R_k)^{-1}, \qquad (2.2)$$

где $\tilde{P}_k$ определяется рекурсивно:

$$\tilde{P}_k = F_{k-1}\hat{P}_{k-1}F_{k-1}^T + G_{k-1}Q_{k-1}G_{k-1}^T, \hat{P}_k = (I - K_k H_k)\tilde{P}_k,$$

тогда $tr\big(Cov[X_k - \hat{X}_k]\big) = tr\big(\hat{P}_k\big) \to min.$

То есть оценка (2.1) будет оптимальной, когда матрица K_k определяется по формуле (2.2). Данный способ оценки вектора состояния называется «фильтром Кальмана». Заметим, что для работы фильтра Кальмана используются матрицы R_k и Q_k. Вопрос их описания и оценки рассматривается отдельно.

Фильтр Кальмана дает оптимальную оценку, когда состояние объекта меняется по закону (1.1), но в общем случае изменение состояния объекта задается по данной формуле:

$$X_{k+1} = F_k X_k + G_k W_k + u_k, \qquad (2.3)$$

где вектор u_k – вектор управления, отвечающий за маневрируемость объекта. Следовательно, фильтр Кальмана дает оптимальную оценку в том случае, когда $u_k = 0$, в противном случае фильтр Кальмана не дает оптимальной оценки. Вопрос о получении оптимальной оценки при $u_k \neq 0$ рассматривается отдельно, в частности, с использованием машинного обучения.

3. Несмещенный фильтр с конечной импульсной характеристикой

Рассмотрим состояние объекта, исходя из полученных измерений, с использованием несмещенного фильтра с конечной импульсной характеристикой, определенной в работе [6].

$$\hat{Y}_k = \tilde{Y}_k + J_k H_k^T \big(\check{X}_k - H_k \tilde{Y}_k\big), \qquad (3.1)$$

где $\tilde{Y}_k = F_{k-1}\hat{Y}_{k-1}$ и J_k определяется по формуле:

$$J_k = [H_k^T H_k + (F_k J_{k-1} F_k)^{-1}]^{-1}. \quad (3.2)$$

Матрица J_k не обновляется после N_{opt}-итераций. Как было показано в работе [6], оценка (3.1) является несмещенной, то есть $M[\hat{Y}_k] = M[X_k]$, но она не обладает свойством оптимальности, которой обладает оценка (2.1) с матрицей K_k, определенной по формуле (2.2). Однако для вычисления $\hat{Y}_k$ не требуются матрицы Q_k и R_k. Кроме того, при рассмотрении модели (2.3) данная оценка дает более точную оценку, чем про использовании фильтра Кальмана при $u_k \neq 0$.

Таким образом, заключение следующее: при $u_k = 0$ стоит выбрать фильтр Кальмана как основную оценку, если же $u_k \neq 0$, то фильтр с конечной импульсной характеристикой будет более правильным выбором. Если u_k был бы известен, то данный вопрос решился бы непосредственно, в этой работе рассматривается случай, когда u_k неизвестно, а оценки матриц Q_k и R_k неточны (как это обычно бывает на практике).

4. Гибрид фильтра Кальмана и несмещенного фильтра с конечной импульсной характеристикой

Данный вопрос о выборе между оценками (2.1) и (3.1) приводит нас к созданию их гибрида, определенным их выпуклой линейной комбинацией.

$$\hat{\Gamma}_k = \alpha_k \hat{X}_k + (1 - \alpha_k)\hat{Y}_k, \quad (4.1)$$

где $\alpha_k \in [0, 1]$. О том, как определить α, далее пойдет и речь.

Обозначим $p_k := P(||\hat{X}_k - X_k|| \leq ||\hat{Y}_k - X_k||)$. Если для $\hat{X}_k$ и $\hat{Y}_k ||\hat{X}_k - X_k|| \leq ||\hat{Y}_k - X_k||$, то, выбрав $\hat{\Gamma}_k$ как основную оценку, мы бы ошиблись на $||\hat{X}_k - \hat{\Gamma}_k||$, в противном случае ошибка представлялась

бы в виде $||\hat{Y}_k - \hat{\Gamma}_k||$. Будем рассматривать данную задачу с точки зрения минимизации максимальной вероятностной ошибки, то есть

$$\max\left(p_k\left\|\hat{X}_k - \hat{\Gamma}_k\right\|, (1 - p_k)\left\|\hat{Y}_k - \hat{\Gamma}_k\right\|\right) \to min.$$

Подставив (4.1) в полученное выражение, имеем

$$\max\left(p_k(1-\alpha_k)\left\|\hat{X}_k + \hat{Y}_k\right\|, (1 - p_k)\alpha_k\left\|\hat{X}_k + \hat{Y}_k\right\|\right) \to min.$$

Данная задача минимизации эквивалентна задаче

$$\max\left(p_k(1-\alpha_k), (1 - p_k)\alpha_k\right) \to min. \tag{4.2}$$

Легко заметить, что при $\alpha_k = p_k$ вышеуказанное выражение будет принимать свое наименьшее значение. Таким образом, чтобы вычислить гибридную оценку надо в первую очередь оценить p_k. Так как оценки $\hat{X}_k$ и $\hat{Y}_k$ являются несмещенными, можем предполагать, что $||\hat{X}_k - X_k||^2 = x^2$ и $||\hat{Y}_k - X_k||^2 = y^2$, где $x \sim N(0,\sigma_1)$ и $y \sim N(0,\sigma_2)$. Оценим вероятность $P(x^2 \le y^2)$.

$$P(x^2 \le y^2) = \frac{1}{2\pi\,\sigma_1\sigma_2}\iint\limits_{x^2\le y^2} e^{-\frac{x^2}{2\sigma_1^2}} e^{-\frac{y^2}{2\sigma_2^2}}dxdy$$

$$= \frac{1}{2\pi\,\sigma_1\sigma_2}\int\limits_{-\infty}^{+\infty}\int\limits_{-|y|}^{|y|} e^{-\left(\frac{x}{\sqrt{2}\sigma_1}\right)^2} e^{-\left(\frac{y}{\sqrt{2}\sigma_2}\right)^2} dxdy$$

$$= \frac{4}{\pi}\int\limits_0^{+\infty}\int\limits_0^{\frac{\sigma_2}{\sigma_1}\eta} e^{-(\xi^2+\eta^2)}d\xi d\eta = \frac{4}{\pi}\int\limits_0^{+\infty}\int\limits_0^{arctg\left(\frac{\sigma_2}{\sigma_1}\right)} e^{-\rho^2}\rho d\theta d\rho$$

$$= \frac{2}{\pi}arctg\left(\frac{\sigma_2}{\sigma_1}\right)\int\limits_0^{+\infty} e^{-\rho^2}d\rho^2 = \frac{2}{\pi}arctg\left(\frac{\sigma_2}{\sigma_1}\right).$$

Таким образом, для того чтобы минимизировать (4.2), необходимо определить α_k, равное $arctg\left(\frac{\sigma_2}{\sigma_1}\right)$. Для этого необходимо оценить σ_1 и σ_2. Это можно произвести эвристическим образом на основе последних l-измерений:

$$\hat{\sigma}_1 = \sqrt{\frac{1}{l}\sum_{i=k-l+1}^{k} ||H\hat{X}_i - Z_i||^2}, \quad \hat{\sigma}_2 = \sqrt{\frac{1}{l}\sum_{i=k-l+1}^{k} ||\hat{Y}_i - Z_i||^2}.$$

Заметим, что даже если $\hat{\sigma}_1$ и $\hat{\sigma}_2$ будут смещенными оценками, то $\frac{\hat{\sigma}_2}{\hat{\sigma}_1}$ будет несмещенной оценкой для $\frac{\sigma_2}{\sigma_1}$.

5. Анализ эффективностиподходов и заключение

Ниже приведены наблюдения ошибок $||\hat{X}_k - X_k||$, $||\hat{Y}_k - X_k||$ и $\left\|\hat{\Gamma}_k - X_k\right\|$, соответственно, для фильтра Кальмана, Несмещенного фильтра с конечной импульсной характеристикой (НФКИХ) и гибридного фильтра при модели, рассмотренной в работе [4, стр. 48–49]:

$$X_k = \begin{bmatrix} x_k \\ y_k \\ \dot{x}_k \\ \dot{y}_k \end{bmatrix}, \qquad F_k = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad G_k = \begin{bmatrix} \frac{T^2}{2} & 0 \\ 0 & \frac{T^2}{2} \\ T & 0 \\ 0 & T \end{bmatrix},$$

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}.$$

С примером ковариационных матриц $Q_k = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$ и $R_k = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}$.

Для фильтра Кальмана оценки матриц были вычислены, как

$$\hat{Q}_k = \begin{bmatrix} 0.8261 & 0.029 \\ 0.029 & 0.9238 \end{bmatrix} \text{ и } \hat{R}_k = \begin{bmatrix} 4.928 & 0.0012 \\ 0.0012 & 5.0289 \end{bmatrix}.$$

А для НФКИХ N_{opt} было взято, равным 8.

Было рассмотрено два возможных случая – на Рис. 1 была рассмотрена модель без маневров (т.е. $u_k = 0$ при любом k), а на Рис. 2 было произведено три случайных маневра. На обоих рисунках координата x – номер измерения k, а y – ошибка соответствующего фильтра.

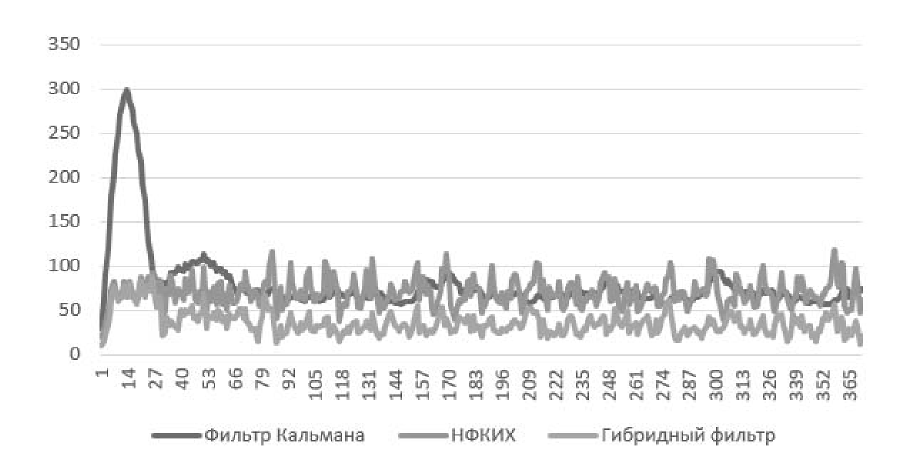


Рис. 1. Статичный процесс (без маневров).



Рис. 2. Процесс со случайными маневрами.

Определим также для трехмерной модели, рассмотренной в работе [4, стр. 64].

$$X_k = \begin{bmatrix} x_k \\ y_k \\ z_k \\ \dot{x}_k \\ \dot{y}_k \\ \dot{z}_k \end{bmatrix}, \quad F_k = \begin{bmatrix} 1 & 0 & 0 & T & 0 & 0 \\ 0 & 1 & 0 & 0 & T & 0 \\ 0 & 0 & 1 & 0 & 0 & T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \quad G_k = \begin{bmatrix} \frac{T^2}{2} & 0 & 0 \\ 0 & \frac{T^2}{2} & 0 \\ 0 & 0 & \frac{T^2}{2} \\ T & 0 & 0 \\ 0 & T & 0 \\ 0 & 0 & T \end{bmatrix},$$

$$H_k = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}.$$

С примером ковариационных матриц

$$Q_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ и } R_k = \begin{bmatrix} 5 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 5 \end{bmatrix}.$$

Для фильтра Кальмана оценки матриц были вычислены, как

$$\hat{Q}_k = \begin{bmatrix} 0.8261 & 0.029 & 0.0148 \\ 0.029 & 0.9238 & 0.0199 \\ 0.0148 & 0.0199 & 0.918 \end{bmatrix} \text{ и } \hat{R}_k = \begin{bmatrix} 4.928 & 0.0012 & 0.0005 \\ 0.0012 & 5.0289 & 0.031 \\ 0.0005 & 0.031 & 4.9917 \end{bmatrix}.$$

А для НФКИХ N_{opt} было взято, равным 8.

Как и в предыдущем примере, было рассмотрено два возможных случая – на Рис. 3 была рассмотрена модель без маневров (т.е. $u_k = 0$ при любом k), а на Рис. 4 было произведено три случайных маневра.

Как можем видеть, при практическом наблюдении гибридный фильтр проявляет себя лучше, чем оба других фильтра во всех случаях. Значимость данной работы заключается в том, что решаются сразу два вопроса – точность оценок $\hat{Q}_k$ и $\hat{R}_k$ и распознование маневров.



Рис. 3. Статичный процесс (без маневров).

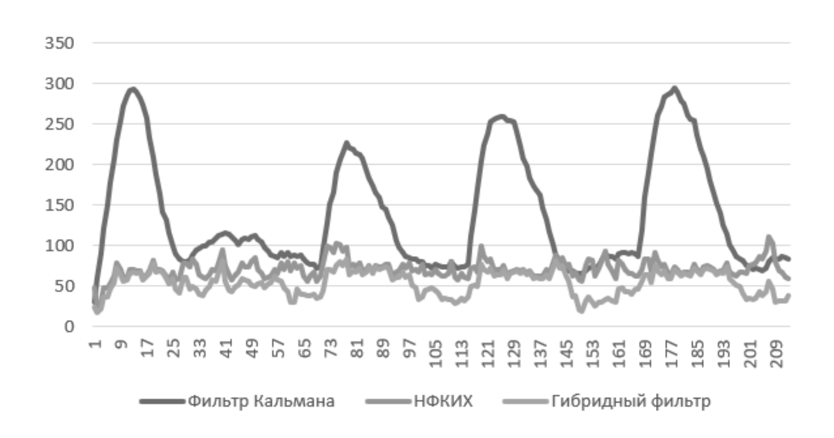


Рис. 4. Процесс со случайными маневрами.

ЛИТЕРАТУРА

1. *Kalman R.E.* (1960). A new approach to linear filtering and prediction problems // Journal of Basic Engineering. 82 (1): 35.45. doi:10.1115/1.3662552. Archived from the original (PDF) on 2008-05-29. Retrieved 2008-05-03.
2. *Kalman R.E.; Bucy R.S.* (1961). New Results in Linear Filtering and Prediction Theory.
3. *Hakobyan Y.R.* Basics of Numerical Analysis (2005).

4. *Ramachandra K.V.* (2000) Kalman Filtering Techniques for Radar Tracking 1st Edition.
5. *Дарбинян А.А., Акопян А.Р.* (2019) Модификация фильтра Калмана для полярных и сферических систем координат// «Вестник РАУ».
6. Unbiased FIR Filtering: An Iterative Alternative to Kalman Filtering Ignoring Noise and Initial Conditions. October 2017IEEE Control Systems Magazine 37(5) DOI: 10.1109/MCS.2017.2718830.
7. Sadhana Vol. 41, No. 12, December 2016, PP. 1491–1507 // "Indian Academy of Sciences DOI 10.1007/s12046-016-0563-y A heuristic reference recursive recipe for adaptively tuning the Kalman filter statistics part-2: real data studies".
8. *Vachhani P., Rengaswamy R., Gangwal V., Narasimhan S.* Recursive estimation in constrained nonlinear dynamical systems', AIChE J., 2005, 51, (3), PP. 946–959.
9. *Ko S., Bitmead R.* 'State estimation for linear systems with state equality constraints' // Automatica, 2007, 43, (8). PP. 1363–1368.
10. *Chia T.* 'Parameter identification and state estimation of constrained systems'. PhD thesis, Case Western Reserve University, 1985.
11. *Rhodes I.* 'A tutorial introduction to estimation and filtering', IEEE Trans. Autom. Control, 1971, AC-16, (6). PP. 688–706.

ABOUT ONE HYBRID SCHEME FORUNBIASEDFINITE IMPULSERESPONSE FILTERAND KALMAN FILTER

A. Darbinyan, A. Hakobyan, M. Khachaturyan

Russian-Armenian University

ABSTRACT

In radio location optimization problem saquestion of ten risesbet ween choosinga finite impulse response filter and Kalman filter. Both these filters have their own weak and strong features. Inthisarticle, for gaining a more optimal estimate for the object position, a hybrid filter was constructed based on the both above mentioned filters.

Keywords: Kalman Filter, Impulse Filter, Hybrid Filter.

УДК 519 218.5

Поступила: 12.02.2021г.
Сдана на рецензию: 12.02.2021 г.
Подписана к печати: 19.02.2021г.

CONVEXITY OF THE DISTRIBUTION OF THE DISTANCE OF TWO RANDOM POINTS IN A CONVEX DOMAIN IN R^2

R. Aramyan, V. Mnatsakanyan

Russian-Armenian University

rafikaramyan@yahoo.com,v.mnatsakanyan95@gmail.com

ABSTRACT

The paper deals with the classical problems of stochastic tomography: obtaining information about a convex domain from the distribution of characteristics of its k-dimensional sections. In this paper, we show that the density function of the squared distance between two uniformly distributed random points in a convex domain is convex.

Subclass: 53C65, 53C60, 31A10

Keywords: convex set, conditional distribution, uniformly distributed random points, random chord.

1. Introduction

Today, tomography is one of the rapidly developing areas of mathematics. Geometric tomography (the term introduced by R. Gardner in [4], see also [3]) is a field of mathematics engaged in extracting information about a geometric object from data on its sections or projections to reconstruct the geometric object. The main tasks of stochastic tomography

are to reconstruct a convex domain by the distribution of its random k-dimensional sections, or projections of the body. The stochastic tomography addresses the following problems: using the concepts known in stochastic geometry, receives new representations for body characteristics. Also, using the data of these concepts in some cases to reconstruct the shape of the body by proposing reconstruction algorithms. The reconstruction of convex bodies by random sections makes it possible to simplify the calculation since mathematical statistics methods can be used to estimate the geometric characteristics of random sections. The integral geometric concepts such as the distribution of the distance between two random points in a convex domain D, carry some information about *D*. In this article, we show that the density function of the squared distance between two uniformly distributed random points in a convex domain is convex.

By R^2 we denote the two-dimensional Euclidean space.

Definition 1. A domain $B \epsilon R^2$is called almost convex if there exists a convex domain $D \epsilon R^2$, that

$$L_2(B\Delta D) = 0$$

here $B\Delta D = (B/D) \cup (D/B)$ is the symmetric difference of B and D, L_2 is the Lebesgue'smeasure in R^2.

For two independent uniformly distributed points P_1, P_2 in a convex domain D, by $r = |P_1 - P_2|$ we denote the distance between the points.

Definition 2. The cumulative distribution function of the distance *r* between two uniformly distributed random points in a convex domain *D* defined by the formula:

$$F_r(u) = \frac{1}{S^2} \int_{\{|P_1 - P_2| < u, P_1, P_2 \in D\}} dP_1 \, dP_2, \tag{1.2}$$

Here S is the area of D, dP_i $(i = 1,2)$ is the element of Lebesgue's measure in R^2.

The cumulative distribution function of $\mathbf{r}^2$defined by the formula

$$F_{r^2}(u) = \frac{1}{S^2}\int_{\{|P_1-P_2|^2<u, P_1,P_2 \in D\}} dP_1\, dP_2 \tag{1.3}$$

Note, that integrals (1.2) and (1.3) do not change if instead of almost convex domain *B* we take the corresponding convex domain *D*. That is why below we will prove the theorems for convex domains.

By $f_{r^2}(u)$ we denote the density function of the squared distance between two uniformly distributed random points P_1,P_2 in a convex domain$D$.The main result of this article is the following theorem.

Theorem 1. Let $D\epsilon R^2$ be an almost convex domain. The density function $f_{r^2}(u)$ of the squared distance of two independent and uniformly distributed points in *D* is decreasing and convex.

Also, we have the following consequence of Theorem 1.

Theorem 2. Let $D\epsilon R^2$ be an almost convex domain. For k ≥ 2 the density function $f_{r^k}(u)$ of $\boldsymbol{r^k}$, where r isthe distance of two independent and uniformly distributed points in *D* is decreasing and convex.

2. Two equivalent representations of the pair of points

By we denote the unit circle in centered at the origin. We denote by the space of lines in the plane. We use the usual parameterization of a line, p is the distance of from the origin is the direction normal to. It is well known that the invariant measure can be decomposed (see [1]).

$$dg = d\varphi dp \tag{1.1}$$

where is the element of the arc measure on.

We consider a pair of points in the plane. There are two equivalent representations of the pair.

1. A pair of points can be determined by the usual Cartesian coordinates

$$(P_1, P_2) = (x_1, y_1, x_2, y_2) \tag{2.1}$$

2. A pair of points can be determined by the line passing through the points and two one-dimensional coordinates, which determine and on the line. Thus

$$(P_1, P_2) = (g, t_1, t_2) = (p, \varphi, t_1, t_2) \tag{2.2}$$

Note that as a reference point on g we take the perpendicular of the origin Oonto. The transform can be represented by the following system

$$\begin{cases} x_i = pcos\varphi - t_i sin\varphi \\ y_i = psin\varphi - t_i cos\varphi \end{cases} \tag{2.3}$$

Easy to calculate that the Jacobian of the transform is (see [2]).

$$dP_1 dP_2 = |t_1 - t_2| dp d\varphi dt_1 dt_2 = |t_1 - t_2| dg dt_1 dt_2 \tag{2.4}$$

3. Proof of Theorem 1.

For a convex domain DϵR^2 and u ∈ [0, ∞) taking into account (2.4) we have

$$F_r(u) = \frac{1}{S^2} \int_{\{|P_1 - P_2| < u; ,P_1, P_2 \in D\}} dP_1\, dP_2 =$$
$$\frac{1}{S^2} \int_{\{g \cap D \neq \emptyset\}} \left[\int_{|\{t_1 - t_2| < u,; t_1, t_2 \in \chi(g)\}} |t_1 - t_2|\, dt_1 dt_2 \right] \text{dg}, \tag{3.1}$$

where t_1 and t_2 are two points chosen at random, independently and with uniform distribution in the segment χ(g)= $g \cap D$. We need to prove the following Lemma.

Lemma 1. For $u \leq X = |\chi(g)|$ we have

$$\int_{\{|t_1 - t_2| < u;\, t_1, t_2 \in \chi(g)\}} |t_1 - t_2|\, dt_1 dt_2 = \left(Xu^2 - \frac{2u^3}{3} \right) \tag{3.2}$$

and for $u > X = |\chi(g)|$ we have

$$\int_{\{|t_1-t_2|<u;\; t_1,t_2\in\chi(g)\}} |t_1-t_2|\, dt_1 dt_2 = \frac{X^3}{3}. \qquad (3.3)$$

Proof of Lemma 1. By symmetry we have: for $u > X$

$$\int_{\{|t_1-t_2|<u;\; t_1,t_2\in\chi(g)\}} |t_1-t_2|\, dt_1 dt_2 = 2\int_0^x d\, t_1 \int_0^{t_1} (t_1-t_2)\, d\, t_2 = \frac{X^3}{3}.$$

For $u \le X$

$$\int_{\{|t_1-t_2|<u;\; t_1,t_2\in\chi(g)\}} |t_1-t_2|\, dt_1 dt_2 = \int_{\{|t_1-t_2|<u;\; t_1,t_2\in\chi(g)\}} |t_1-t_2|\, dt_1 dt_2 - \int_{\{|t_1-t_2|\ge u;\; t_1,t_2\in\chi(g)\}} |t_1-t_2|\, dt_1 dt_2 = \frac{X^3}{3} - 2\int_u^X dt_1 \int_0^{t_1-u} (t_1-t_2)\, dt_2 = Xu^2 - \frac{2u^3}{3}.$$

Lemma 1 is proved.

Substituting (3.2) and (3.3) into (3.1) we obtain

$$F_r(u) = \frac{1}{S^2}\int_{\{g:\, g\cap D\neq\emptyset\}} \left[I_{\{u\le x\}} \left(Xu^2 - \frac{2u^3}{3} \right) + I_{\{u>x\}} \frac{X^3}{3} \right] dg, \qquad (3.4)$$

here $I_{\{A\}}$ is the indicator function of *A*. From (3.4) for the cumulative distribution function of $\mathbf{r}^2$ we have

$$F_{\mathbf{r}2}(u) = F_r(\sqrt{u}) = \frac{1}{S^2}\int_{\{g:\, g\cap D\neq\emptyset\}} [I_{\{u\le X^2\}} \left(Xu \quad - \frac{2u^{3/2}}{3} \right) + I_{\{u>X^2\}} \frac{X^3}{3}]\ dg. \qquad (3.5)$$

For the density function $f_{r^2}(u)$ of squared distance between two uniformly distributed points P_1, P_2 in a convex domain D, we must differentiate under the integral sign in (3.5). For the derivative for u≠X^2 we get (for u=X^2 we assume that the derivative is 0)

$$f_{\mathbf{r}^2}(u) = \frac{1}{S^2}\int_{\{g:\, g\cap D\neq\emptyset\}} \left[I_{\{u\le X^2\}} \left(Xu \quad - \frac{2u^{\frac{3}{2}}}{3} \right) + I_{\{u>X^2\}} \frac{X^3}{3} \right]' dg = \frac{1}{S^2}\int_{\{g:\, g\cap D\neq\emptyset\}} \left[I_{\{u\le X^2\}} (X-\sqrt{u}) + 0 \right] \quad dg. \qquad (3.6)$$

It is obvious that the function under the integral sign in (3.6) is convex on the interval $[0,X^2]$ and the integration with respect to a positive measure is convex. Theorem 1 is proved.

4. An another proof of Theorem 1

Note that in [5] was found the following relation between the distribution function of the distance of two uniformly distributed points of D and the chord length distribution function of D.

Theorem 3. Let D be a convex domain and $F_r(u)$ *is* the cumulative distribution function of the distance of two independent uniformly distributed points in D, then

$$F_r(u) = 1 - \frac{L}{S^2}\left(\frac{2u^3}{3} - \frac{\pi S u^2}{L} + \frac{S^2}{l} - u^2\int_0^u F_X(v)\,dv + \int_0^u v^2 F_X(v)\,dv\right) \quad (4.1)$$

here $F_X(u)$random chord length cumulative distribution function of D.

From here we can get the cumulative distribution function of the square of the distance of two independent and uniformly distributed points in D

$$F_{r^2}(u) = 1 - \frac{L}{S^2}\left(\frac{2u^{1.5}}{3} - \frac{\pi S u}{L} + \frac{S^2}{L} - u\int_0^{\sqrt{u}} F_X(v)\,dv + \int_0^{\sqrt{u}} v^2 F_X(v)\,dv\right) \quad (4.2)$$

By calculating derivative of this function, we will get density function

$$f_{r^2}(u) = -\frac{L}{S^2}\left(\sqrt{u} - \pi S - \int_0^{\sqrt{u}} F_X(v)\,dv\right). \quad (4.3)$$

By calculating the first and the second derivative of density function we will get

$$f'_{r^2}(u) = -\frac{L}{2S^2}\left(\frac{1-F_X(\sqrt{u})}{\sqrt{u}}\right) = \frac{L}{2S^2}\left(\frac{F_X(\sqrt{u})-1}{\sqrt{u}}\right) \quad (4.4)$$

$$f_{r^2}''(u) = \frac{L}{2S^2}\left(\frac{f_X(\sqrt{u})\sqrt{u}+1-F_X(\sqrt{u})}{2u\sqrt{u}}\right) = \frac{L}{2S^2}\left(\frac{f_X(\sqrt{u})}{2u} + \frac{1-F_X(\sqrt{u})}{2u\sqrt{u}}\right) \qquad (4.5)$$

It is obvious that the first derivative is negative and the second one is positive. Hence our function is decreasing and convex.

5. Proof of Theorem 2

Lemma 2. Let the derivative of $F(x)$ be a decreasing non-negative convex function defend on a segment than the derivative of $\mathrm{G(x)} = F(x^k)$, $0 < k < 1$ is also decreasing and convex.

Proof of Lemma 2.

For the derivative of G(x) we will get

$$G'(x) = kf(x^k)x^{k-1} = g(x) \qquad (5.1)$$

and will calculate the first and the second derivative of $g(x)$

$$g'(x) = k^2 f'(x^k)x^{2k-2} + k(k-1)f(x^k)x^{k-2} \qquad (5.2)$$

$$g''(x) = k^3 f''(x^k)x^{2k-2} + (2k-2)k^2 f'(x^k)x^{2k-3} + k^2(k-1)f'^{(x^k)x^{k-2}} + k(k-1)(k-2)f(x^k)x^{k-3} \qquad (5.3)$$

obvious that the first derivative of $g(x)$ is negative and the second derivative is positive.

Corollary. From this lemma it is obvious if theorem 1 is true, consequently, theorem 2 is also true.

Note, that the convers statement: whether or not a convex domain D is convex if the density function $f_{r^2}(u)$ of the square distance of two independent and uniformly distributed points in D is convex remains open.

REFERENCES

1. *Ambartzumian R.V.* Combinatorial integral geometry, metric and zonoids. Acta Appl. Math. 1987, vol. 9, 3–27.
2. *Santalo L.A.* Integral Geometry and Geometric Probability, Addison-Wesley, Reading, Massachusetts, USA (1976).
3. *Blaschke W., Kreis und Kugel* (in German), 2nd Ed., W. de Gruyter, Berlin, 1956.
4. Gardner R.J. Geometric tomography, Second ed., Cambridge University Press, Cambridge, 2006.
5. *Aramyan R.H., Mnatsakanyan V.A.* Conditional moments of the distance distribution two random points in a convex domain in R^2, Proceedings of YSU, Physical and Mathematical Sciences, 2020, v. 54, #1. PP. 3–8.

ВЫПУКЛОСТЬ РАСПРЕДЕЛЕНИЯ РАССТОЯНИЯ ДВУХ СЛУЧАЙНЫХ ТОЧЕК В ВЫПУКЛОЙ ОБЛАСТИ В R^2

Р. Арамян, В. Мнацаканян

Российско-Армянский Университет

АННОТАЦИЯ

В данной статье рассматриваются классические задачи стохастической томографии: получить информацию о выпуклой области из распределения характеристик ее k-мерных сечений. В данной статье показывается, что функция плотности квадрата расстояния между двумя равномерно распределенными случайными точками в выпуклой области является выпуклой.

Подкласс: 53C65, 53C60, 31A10

Ключевые слова: выпуклое множество, условное распределение, равномерно распределенные случайные точки, случайная хорда.

УДК: 517.968.4

Поступила: 22.12.2020г.
Сдана на рецензию: 25.12.2021г.
Подписана к печати: 15.01.2021г.

О РАЗРЕШИМОСТИ ОДНОЙ СИСТЕМЫ НЕЛИНЕЙНЫХ ИНТЕГРАЛЬНЫХ УРАВНЕНИЙ НА ПОЛУОСИ

А.К. Кроян

Армянский национальный аграрный университет

arpi.kroyan@mail.ru

АННОТАЦИЯ

Данная научная статья посвящена изучению и решению одной системы интегральных уравнений с монотонной и выпуклой нелинейностью на положительной части числовой прямой. Разные частные случаи исследуемых систем имеют непосредственные применения в теории p-адической струны, в математической биологии и в кинетической теории газов. При определенных ограничениях на нелинейность и на ядро исследуемой системы доказана теорема существования покомпонентно неотрицательных нетривиальных и ограниченных решений. Изучено также асимптотическое поведение построенного решения на бесконечности. Приведены конкретные примеры указанных систем для иллюстрации полученных результатов.

Библиография: 8 наименований.

Ключевые слова: система интегральных уравнений, нелинейность, итерации, монотонность, сходимость.

1. Введение и формулировка основной теоремы

Рассмотрим следующую систему нелинейных интегральных уравнений:

$$Q_i\big(f_i(x)\big) = \sum_{j=1}^{n} \int_0^{\infty} K_{ij}(x,t)f_j(t)\,dt, \qquad x \in \mathbb{R}^+, \qquad i = 1,2,\dots,n \tag{1}$$

относительно искомой измеримой вектор-функции

$$f(x) := \big(f_1(x), f_2(x), \dots, f_n(x)\big)^T$$

(T – знак транспонирования).

Матричное ядро $K(x,t) := (K_{ij}(x,t))_{i,j=1}^{n\times n}$ – определенная на множестве $\mathbb{R}^+ \times \mathbb{R}^+ (\mathbb{R}^+ := [0,+\infty))$ существенно ограниченная функция, удовлетворяющая следующим условиям:

1) $r(A) = 1$, $r(A)$ – спектральный радиус матрицы $A = (a_{ij})_{i,j=1}^{n\times n}$:

$$0 < a_{ij} := \sup_{x\geq 0} \int_0^{\infty} K_{ij}(x,t)dt, \quad i,j = 1,2,\dots,n, \tag{2}$$

2) существуют функции $\{\widetilde{K}_{ij}(x)\}_{ij=1}^{n\times n}$, $\{\lambda_j(t)\}_{j=1}^{n}$ и число $\varepsilon_0 > 0$, удовлетворяющие следующим условиям:

2a) $\widetilde{K}_{ij}(-x) = \widetilde{K}_{ij}(x)$, $x \geq 0$, $0 < \widetilde{K}_{ij} \in L_1(\mathbb{R}) \cap L_\infty(\mathbb{R})$,

$$\int_{-\infty}^{\infty} \widetilde{K}_{ij}(x)dx = a_{ij}, \qquad i,j = 1,2,\dots,n, \tag{3}$$

($L_\infty(\mathbb{R})$ – пространство существенно ограниченных на $\mathbb{R}$-функций);

2b) $\widetilde{K}_{ij}(x) \downarrow$ по x на $\mathbb{R}^+$,

$$\int_0^{\infty} x\widetilde{K}_{ij}(x)dx < +\infty, \qquad i,j = 1,2,\dots,n,$$

$$\varepsilon_0 \leq \lambda_j(t) \leq 1, \quad 1-\lambda_j \in L_1(\mathbb{R}^+), \quad \lambda_j(t) \uparrow \text{ по } t,$$

$$\lim_{t\to\infty} \lambda_j(t) = 1, \quad j = 1,2,\dots,n,$$

такие, что

$$K_{ij}(x,t) \geq \big(\widetilde{K}_{ij}(x-t) - \widetilde{K}_{ij}(x+t)\big)\lambda_j(t),$$
$$(x,t) \in \mathbb{R}^+ \times \mathbb{R}^+, \qquad i,j = 1,2,\dots,n. \tag{4}$$

Из условия 1) в силу теоремы Перрона (см. [1]) следует, что существует вектор $\eta = (\eta_1, \eta_2, \dots, \eta_n)^T$ с положительными координатами $\{\eta_i\}_{i=1}^n$, для которого

$$\sum_{j=1}^{n} a_{ij}\,\eta_j = \eta_i\,, \quad i = 1,2,\dots,n. \tag{5}$$

Введем обозначение

$$\eta_i^* := \frac{\eta_i}{\min\limits_{1\le i\le n} \eta_i}\ , \ \ i = 1,2,\dots,n. \tag{6}$$

Будем считать, что функции $\{Q_i(u)\}_{i=1}^n$ удовлетворяют следующим условиям:

A) $Q_i \in \mathbb{C}[0,\eta_i^*], Q_i(u) \uparrow$ по u на $[0,\eta_i^*]$, $i = 1,2,\dots,n,$

B) $Q_i(u)$ – выпуклые вниз функции на отрезке $[0,\eta_i^*]$, $i = 1,2,\dots,n,$

C) $Q_i(\eta_i^*) = \eta_i^*,$ $Q_i(0) = 0$ и функциональные уравнения $Q_i(u) = \varepsilon_0 u$ имеют положительные решения ξ_i, причем $0 < \xi_i < \eta_i^*, i = 1,2,\dots,n.$

Система (1) для различных видов ядер K_{ij} и нелинейностей Q_i (удовлетворяющих, соответственно, условиям 1) - 2), 2a) -2c) и A) – C)) имеет приложения в теории p-адических открыто-замкнутых струн, в математической теории пространственно-временного распространения эпидемии и в кинетической теории газов (см.[3–8]).

При $\lambda_j(t) \equiv 1,\ j = 1,2,\dots,n,$ когда в (4) имеет место знак равенства, система (1) исследована в работе Х.А. Хачатряна (см. [4]). Соответствующее скалярное интегральное уравнение (1) при аналогичных ограничениях на ядро и нелинейность достаточно подробно было исследовано в работе (см.[5]).

В настоящей работе при условиях 1) – 2), 2a) – 2c), A) – C) для общей системы (1) доказывается существование покомпонентно неотрицательного нетривиального и ограниченного решения. Более то-

го, изучается также асимптотическое поведение построенного решения на бесконечности. В заключении работы приводятся частные примеры рассматриваемых систем, имеющие прикладной характер. Основным результатом настоящей статьи является следующая теорема.

Теорема. *При условиях 1) – 2), 2a) – 2c) и A) – C) система (1) имеет покомпонентно неотрицательное нетривиальное и существенно ограниченное решение* $f(x):=\big(f_1(x),\ f_2(x),\dots,f_n(x)\big)^T$, *причем*

$$0\le f_i(x)\ \le \eta_i^*,\ x\in\mathbb{R}^+,$$

$$\lim_{x\to\infty} f_i(x)=\ \eta_i^*,\ \eta_i^*-f_i\in L_1(\mathbb{R}^+),\quad i=1,2,\dots,n\,.$$

2. Доказательство теоремы

Рассмотрим следующую систему нелинейных интегральных уравнений

$$Q_i\big(\varphi_i(x)\big)=\sum_{j=1}^{n}\int_0^\infty\Big(\widetilde{K}_{ij}(x-t\,)-\widetilde{K}_{ij}(x+t\,)\Big)\,\varphi_j(t)dt,$$
$$x\in\mathbb{R}^+,\ \ i=1,2,\dots,n \qquad (7)$$

относительно искомой вектор-функции

$$\varphi(x):=\big(\varphi_1(x),\varphi_2(x),\dots,\varphi_n(x)\big)^T.$$

Согласно результатам работы [4] система (7) имеет покомпонентно неотрицательное нетривиальное ограниченное и непрерывное на $\mathbb{R}^+$ решение $\varphi(x):=\big(\varphi_1(x),\varphi_2(x),\dots,\varphi_n(x)\big)^T$, удовлетворяющее следующим условиям:

$$0\le\varphi_i(x)\ \le\eta_i^*,\qquad x\in\mathbb{R}^+,\ \ \lim_{x\to\infty}\varphi_i(x)=\ \eta_i^*\,, \qquad (8)$$

$$\eta_i^*-\varphi_i\in\ L_1(\mathbb{R}^+),\ \ i=1,2,\dots,n. \qquad (9)$$

Теперь рассмотрим следующую систему нелинейных интегральных уравнений:

$$Q_i\big(\psi_i(x)\big) = \sum_{j=1}^{n} \int_0^\infty \Big(\widetilde{K}_{ij}(x-t\,) - \widetilde{K}_{ij}(x+t\,)\Big)\lambda_j(t)\psi_j(t)dt,$$

$$x \in \mathbb{R}^+, \;\; i = 1,2,\dots,n \qquad (10)$$

относительно искомой вектор-функции

$$\psi(x) := \big(\psi_1(x), \psi_2(x), \dots, \psi_n(x)\big)^T.$$

Для системы (10) рассмотрим следующие итерации:

$$Q_i\left({\psi_i}^{(m+1)}(x)\right) = \sum_{j=1}^{n} \int_0^\infty \Big(\widetilde{K}_{ij}(x-t\,) - \widetilde{K}_{ij}(x+t\,)\Big)\lambda_{,j}\,(t){\psi_j}^{(m)}(t)dt,$$

$${\psi_i}^{(0)}(x) = \eta_i^*, i = 1,2,\dots,n, m = 0,1,2,\dots, x \in \mathbb{R}^+. \qquad (11)$$

Учитывая условие 2c), и монотонность функций $Q_i(u), i = 1,2,\dots,n$ по u, индукцией по m несложно доказать, что

$$0 \le {\psi_i}^{(m)}(x) \;\downarrow\; \text{по } m, i = 1,2,\dots,n. \qquad (12)$$

Ниже докажем, что имеют место оценки

$${\psi_i}^{(m)}(x) \ge \min_{1\le j\le n}\left(\frac{\xi_j}{\eta_j^*}\right)\eta_i^*\big(1-e^{-p^*x}\big),$$

$$i = 1,2,\dots,n\,, m = 0,1,2,\dots, x \in \mathbb{R}^+, \qquad (13)$$

где числа $0 < \xi_i < \eta_i^*$ определяются из условия С),
а $p^* = \min(p_1, p_2, \dots, p_n)$ и числа $\{p_i\}_{i=1}^n$ – единственным образом из следующих характеристических уравнений:

$$\sum_{j=1}^{n} \eta_j^* \int_0^\infty \widetilde{K}_{ij}\,(t)e^{-pt}\,dt = \frac{\varepsilon_0\,\eta_i^*}{2}\,,\;\; i = 1,2,\dots,n. \qquad (14)$$

С этой целью воспользуемся априорной оценкой, доказанной в работе [4].

Неравенство (13) для $m = 0$ непосредственно следует из следующей оценки:

$$\sum_{j=1}^{n} \eta_j^* \int_0^\infty \left(\widetilde{K}_{ij}(x-t) - \widetilde{K}_{ij}(x+t)\right)\left(1-e^{-p^* t}\right)dt \geq$$
$$\geq \varepsilon_0\, \eta_i^*\left(1-e^{-p^* x}\right), \; i = 1,2,\dots,n, \; x \in \mathbb{R}^+. \tag{15}$$

Неравенство (13) для $m = 0$ непосредственно следует из следующей оценки

$$\eta_i^* \geq \eta_i^*\left(1-e^{-p^* x}\right) \geq \min_{1\leq j\leq n}\left(\frac{\xi_j}{\eta_j^*}\right)\eta_i^*\left(1-e^{-p^* x}\right),$$
$$i = 1,2,\dots,n, \; x \in \mathbb{R}^+. \tag{16}$$

Предположим, что неравенство

$$\psi_i{}^{(m)}(x) \geq \min_{1\leq j\leq n}\left(\frac{\xi_j}{\eta_j^*}\right)\eta_i^*\left(1-e^{-p^* x}\right), \quad i = 1,2,\dots,n, \qquad x \in \mathbb{R}^+$$

имеет место при некотором $m \in \mathbb{N}$.Тогда из (11), в силу оценки (15) и A) – C) имеем, что

$$Q_i\left(\psi_i{}^{(m+1)}(x)\right) \geq \min_{1\leq j\leq n}\left(\frac{\xi_j}{\eta_j^*}\right)\eta_i^*\varepsilon_0\left(1-e^{-p^* x}\right) \geq$$
$$\geq Q_i\left(\min_{1\leq j\leq n}\left(\frac{\xi_j}{\eta_j^*}\right)\eta_i^*\left(1-e^{-p^* x}\right)\right).$$

Отсюда, в силу условий A) – C), получаем,что

$$\psi_i{}^{(m+1)}(x) \geq \min_{1\leq j\leq n}\left(\frac{\xi_j}{\eta_j^*}\right)\eta_i^*\left(1-e^{-p^* x}\right), \quad i = 1,2,\dots,n, \qquad x \in \mathbb{R}^+.$$

Итак, в силу доказанного, последовательность вектор-функций $\{\psi^{(m)}(x)\}_{m=0}^{\infty}$, $\psi^{(m)}(x) := \left(\psi_1^{(m)}(x), \psi_2^{(m)}(x), \dots, \psi_n^{(m)}(x)\right)$ имеет поточечный предел при $m \to \infty$:

$$\lim_{m\to\infty} \psi_i^{(m)}(x) := \psi_i(x), \quad i = 1,2,\dots,n, \tag{17}$$

причем предельные вектор-функции $\psi_i(x)$ в силу теоремы Б. Леви (см.[2]) и непрерывности функций$\{Q_i(u)\}_{i=1}^{n}$ удовлетворяют системе (10). Более того, из (12) и (13) следует, что

$$\min_{1\le j\le n}\left(\frac{\xi_j}{\eta_j^*}\right)\eta_i^*\left(1 - e^{-p^*x}\right) \le \psi_i(x) \le \eta_i^*, \qquad i = 1,2,\dots,n, \qquad x \in \mathbb{R}^+. \tag{18}$$

Проводя аналогичные рассуждения, как в работе [4], с учетом условия 2c), можно показать, что

$$\lim_{x\to\infty}\psi_i(x) = \eta_i^*, \quad \eta_i^* - \psi_i \in L_1(\mathbb{R}^+), \quad i = 1, 2, \dots, n, \tag{19}$$

$$\psi_i(x) \le \varphi_i(x), \quad i = 1,2,\dots,n, \qquad x \in \mathbb{R}^+. \tag{20}$$

Для системы (1) введем следующие последовательные приближения:

$$Q_i\left(f_i^{(m+1)}(x)\right) = \sum_{j=1}^{n}\int_0^\infty K_{ij}(x,t) f_j^{(m)}(t)dt, \qquad x \ge 0,$$

$$f_i^{(0)}(x) = \psi_i(x), \quad m = 0,1,2,\dots, i = 1,2,\dots,n. \tag{21}$$

Индукцией по m можно доказать, что

$$f_i^{(m)}(x) \uparrow \text{по } m, \qquad i = 1,2,\dots,n, \tag{22}$$

$$f_i^{(m)}(x) \le \eta_i^*, \quad i = 1,2,\dots,n, m = 0,1,2,\dots, x \in \mathbb{R}^+. \tag{23}$$

Следовательно, последовательность вектор-функций

$$f^{(m)}(x) := \left(f_1^{(m)}(x), f_2^{(m)}(x), \dots, f_n^{(m)}(x)\right)^T, \qquad m = 0,1,2,\dots,$$

имеет поточечный предел при $m \to \infty$:

$$\lim_{m\to\infty} f_i^{(m)}(x) = f_i(x) , \qquad i = 1,2,\dots,n. \qquad (24)$$

Предельная вектор-функция $f(x) := \left(f_1(x),\ f_2(x),\dots,f_n(x)\right)^T$, в силу непрерывности функций $Q_i(u)$ и теоремы Б. Леви, удовлетворяет системе (1). Более того, из (22) и (23) следует, что

$$\psi_i(x) \le f_i(x) \le \eta_i^* , \quad i = 1,2,\dots,n, \qquad x \in \mathbb{R}^+. \qquad (25)$$

Из (19) и (25) непосредственно следует, что

$$\eta_i^* - f_i \in L_1(\mathbb{R}^+),\ \lim_{x\to\infty} f_i(x) = \eta_i^* , i = 1,2,\dots,n. \qquad (26)$$

Теорема доказана.

3. Примеры

В заключении работы приведем несколько частных примеров, указанных систем нелинейных интегральных уравнений, имеющих приложения в динамической теории p-адических открыто-замкнутых струн и в кинетической тории газов.

Примеры $\{Q_i(u)\}_{i=1}^n$:

a) $Q_i(u) = \frac{u^{\mathrm{p}}}{(\eta_i^*)^{p-1}}$, $i = 1,2,\dots,n$, где $p > 2$ – нечетное число, а числа $\{\eta_i^*\}_{i=1}^n$ определяются из формул (5) и (6) ,

b) $Q_i(u) = \mathrm{c_i} \frac{u^{\mathrm{p}}}{(\eta_i^*)^{p-1}} + (1 - c_i)u$, где $c_i \in (0,1)$, $i = 1,2,\dots,n$ – числовые параметры.

Примеры ядерных функций $\left\{\widetilde{K}_{ij}(x)\right\}_{i,j=1}^{n\times n}$ и $\left\{\lambda_j(t)\right\}_{j=1}^n$:

I) $\widetilde{K}_{ij}(x) = \frac{a_{ij}}{\sqrt{\pi}} e^{-x^2},\ x \in \mathbb{R}$,

где $a_{ij} > 0,\ i,j = 1,2,\dots,n$ и спектральный радиус матрицы $\mathrm{A} = \left(a_{ij}\right)_{i,j=1}^{n\times n}$ равен 1,

ll) $\widetilde{K}_{ij}(x) = \int_a^b e^{-|x|s} G_{ij}(s)ds,\ \ x \in \mathbb{R}, i,j = 1,2,\dots,n,$

где $G_{ij}(s) > 0$, $G_{ij} \in \mathbb{C}\,[a,b),\ 0 < a < b \le +\infty$ и спектральный радиус матрицы

$$\mathrm{A} = \left(2\int_a^b \frac{G_{ij}(s)}{s}\,ds\right)_{i,j=1}^{n\times n}$$ равен 1.

1) $\lambda_j(t) = 1 - \delta_j e^{-t},\quad t \ge 0$, где $\delta_j \in (0,1),\ \ j = 1,2,\dots,n$ – числовые параметры,

2) $\lambda_j(t) = 1 - (1-\varepsilon_0)\mathrm{e}^{-q_j t^2},\quad q_j > 0,\quad j = 1,2,\dots,n,$

где число ε_0 определяется в условии 2c).

И наконец, приведем примеры матричного ядра $\{K_{ij}(x,t)\}_{i,j=1}^{n\times n}$, удовлетворяющего условиям доказанной теоремы:

i1) $K_{ij}(x,t) = \left(\widetilde{K}_{ij}(x-t) - \widetilde{K}_{ij}(x+t)\right)\lambda_j(t),\ \ i,j = 1,2,\dots,n,$
$(x,t) \in \mathbb{R}^+ \times \mathbb{R}^+,$

i2) $K_{ij}(x,t) = \widetilde{K}_{ij}(x-t),\quad j,i = 1,2,\dots,n,\ \ (x,t) \in \mathbb{R}^+ \times \mathbb{R}^+,$

i3) $K_{ij}(x,t) = \widetilde{K}_{ij}(x-t) - \widetilde{K}_{ij}(x+t),\ \ j,i = 1,2,\dots,n,$
$(x,t) \in \mathbb{R}^+ \times \mathbb{R}^+.$

Автор выражает благодарность своему научному руководителю д.ф.м. наук, профессору А.Х. Хачатряну за постановку задачи и за полезные советы при выполнении работы.

ЛИТЕРАТУРА

1. *Гантмахер Ф.Р.* Теория матриц. 2-е изд., доп. М.: Наука, 1966, С. 576.
2. *Колмогоров А.Н., Фомин С.В.* Элементы теории функций и функционального анализа. 5-ое изд., М.: Наука, 1981, С. 544.
3. *Хачатрян А.Х., Хачатрян Х.А.* О разрешимости некоторых нелинейных интегральных уравнений в задачах распространения эпидемии. Тр. МИАН, 306, 2019. СС. 287–303.

4. *Хачатрян Х.А.* О разрешимости одной системы нелинейных интегральных уравнений типа Гаммерштейна на прямой. Изв. Сарат. ун-та. Нов. сер. Математика. Механика. Информатика. Т. 19, Вып. 2, 2019. СС. 164–181.
5. *Хачатрян Х.А.* О разрешимости одной граничной задачи в p – адической теории струн. Тр. ММО, 79, № 1, МЦНМО, 2018. СС. 117–132.
6. *Хачатрян Х.А., Терджян Ц.Э., Аветисян М.О.* Однопараметрическое семейство ограниченных решений для одной системы нелинейных интегральных уравнений на всей прямой. Изв. НАН Армении, Математика, 53:4, 2018. СС. 72–86.
7. *Хачатрян Х.А., Петросян А.С.* О разрешимости одного класса нелинейных интегральных уравнений Гаммерштейна Стилтьеса на всей прямой. Тр. МИАН, 308 (2020). СС. 253–264.
8. *Хачатрян А.Х., Хачатрян Х.А.* О разрешимости нелинейного модельного уравнения Больцмана в задаче плоской ударной волны. ТМФ. Т. 189, № 2. 2016. СС. 239–255.

ON THE SOLVABILITY OF ONE SYSTEM OF NONLINEAR INTEGRAL EQUATIONS ON A SEMI-AXIS

A. Kroyan

Armenian National Agrarian University

ABSTRACT

The work is devoted to the study and solution of one system of integral equations with monotone and convex nonlinearity on the positive part of the number line. Various special cases of this system have direct applications in the theory of p-adic string, in mathematical biology, and in the kinetic theory of gases. Under certain restrictions on nonlinearity and on the kernel of the system under study, a theorem on the existence of component-wise nonnegative nontrivial and bounded solutions is proved. The asymptotic behavior of the constructed solution at infinity is also studied. Specific examples of these systems are given to illustrate the obtained results.

Keywords: system of integral equations, nonlinearity, iteration, monotonicity, convergence.

UDC 519.174.7 Поступила: 04.02.2021г.
Сдана на рецензию: 11.02.2021г.
Подписана к печати: 24.02.2021г.

ON THE PALETTE INDEX OF SIERPIŃSKI-LIKE GRAPHS

A.B. Ghazaryan

Department of Informatics and Applied mathematics
Yerevan State University

ghazaryan.aghasi@gmail.com

ABSTRACT

For a proper edge coloring of a graph G, the palette of a vertex $v \in V(G)$ is the set of colors appearing on the edges incident to the vertex v. The palette index $\check{s}(G)$ of G is the minimum number of palettes occurring in G among all proper edge colorings. In this paper, we examine the palette index of Sierpiński-like graphs. Namely, we determine the palette index of Sierpiński graphs $S(n,k)$ where k is even, $k=3$, or $n=2$ and $k \equiv 3 \pmod 4$ and we give an upper and a lower bound for the remaining cases. Additionally, we determine the palette index of graphs $S^{+}(n,k)$ when k is odd or $k \equiv 0 \pmod 4$ and we give an upper and a lower bound for the remaining cases. Moreover, we completely determine the palette index of graphs$S^{++}(n,k)$.

Keywords: palette index, Sierpiński graphs, Sierpiński-like graphs.

Introduction

In this paper we use the standard notations of graph theory which you can find in [13]. Graph coloring problems are one of the most widely studied problems of graph theory. These problems have many applications,

especially in scheduling theory. Chromatic parameters describe graph colorings and the most popular chromatic parameters are the chromatic number and the chromatic index.

In this paper, we study a new chromatic parameter called the *palette index* of a graph, which is introduced in 2014 by Hornák, Kalinowski, Meszka, and Woźniak [6]. For a proper edge coloring of a graph G, the palette of a vertex $v \in V(G)$ is the set of colors assigned to the edges incident to v. The palette index $\check{s}(G)$ of G is the minimum number of distinct palettes occurring in G for all proper edge colorings.

The parameter has mainly been studied for regular graphs. In [6], the authors determined the palette index of complete graphs. Namely,

Theorem 1. *For every integer* $n > 1$*, we have*

$$\check{s}(K_n) = \begin{cases} 1 & \text{if } n \equiv 0 \pmod 2, \\ 3 & \text{if } n \equiv 3 \pmod 4, \\ 4 & \text{if } n \equiv 1 \pmod 4. \end{cases}$$

They also determined the palette index of cubic graphs. Specifically,

$$\check{s}(G) = \begin{cases} 1 & \text{if } G \text{ is of Class 1}, \\ 3 & \text{if } G \text{ is of Class 2 and has a perfect matching}, \\ 4 & \text{if } G \text{ is of Class 2 and has no perfect matching}. \end{cases}$$

In [6], the authors give the following:

Proposition 1. *The palette index of a graph* G *is* 1 *if and only if the graph is regular and Class 1.*

Also they showed that the palette index of a regular graphs of Class 2 is different from 2.

Proposition 1 yields that even determining if a given graph G has a palette index 1 is a NP-complete problem [12].

The palette index of d-regular graphs G of Class 2 satisfy the inequality $3 \leq \check{s}(G) \leq d+1$ [3]. In [3], it is shown that the palette index of 4-regular graphs is in $\{1,3,4,5\}$ and all these values are attained.

There are few results about the palette index of non-regular graphs.

In [5], the authors determined the palette index of the complete bipartite graphs $K_{a,b}$ where $min\{a,b\} \leq 5$.

In [4], the palette index of bipartite graphs is studied. In particular, they determined the palette index of the grid graphs and characterized the class of graphs whose palette index equals the number of vertices.

In [2], the authors give a sharp upper bound of the palette index of tree graphs.

Vizing's edge coloring theorem gives an upper bound of the palette index of a graph G with the maximum degree Δ. Namely, $\check{s}(G) \leq 2^{\Delta+1} - 2$. In [1], the authors succeeded in constructing a family of multigraphs whose palette index grows asymptotically as Δ^2. Although, it is an open question whether there are such graphs without multiple edges.

One of interesting graph families is Sierpińskigraphs $S(n,k)$. Let us define $S(n,k)$ where n and k are positive integers. The vertex set of $S(n,k)$ is the set of all n-tuples of integers $1,\dots,k$, namely, $V(S(n,k)) = \{1,2,\dots,k\}^n$. Two different vertices $u = (u_1,\dots,u_n)$ and $v = (v_1,\dots,v_n)$ are adjacent if and only if there exists an $h \in \{1,\dots,n\}$ such that

1. $u_t = v_t$, for $t = 1,\dots,h-1$
2. $u_h \neq v_h$
3. $u_t = v_h$and$v_t = u_h$ for $t = h+1,\dots,n$.

The vertices $(i,\dots,i)$ are called the *extreme vertices* of $S(n,k)$ where $i = 1,2,\dots,k$. We denote the subgraph $S(n,k)[\{v|v = (i,\dots)\}]$ of $S(n,k)$ by $S_i(n,k)$ where $i = 1,2,\dots,k$. Clearly, $S_i(n+1,k)$ is isomorphic to $S(n,k)$. Consequently, $S(n,k)$ contains k^{n-1} copies of the graph $S(1,k) = K_k$. We call all the edges of $S(n,k)$ *link edges* that do not belong to the above-mentioned K_k.

For the first time, $S(n,k)$ was introduced in [10] while studying Lipscomb space. It is noteworthy that $S(n,3)$ is isomorphic to the graph of

the Tower of Hanoi with n disks [10]. These graphs are widely used in various fields of science [7], [9].

Sierpiński graphs are "almost" regular. There are different ways to change $S(n,k)$ into regular graph. The two quite natural ways are $S^{+}(n,k)$ and $S^{++}(n,k)$. These graphs are introduced in [11] and are defined as follows: Let n and k be positive integers. The graph $S^{+}(n,k)$ is obtained from $S(n,k)$ by adding a vertex w, called *special vertex*, and connects all extreme vertices of $S(n,k)$ to w.

If $n = 1$ then the graph $S^{++}(n,k)$ is isomorphic to K_{k+1}. If $n > 1$ then $S^{++}(n,k)$ is the graph obtained from the disjoint union of $k+1$ copies of $S(n-1,k)$ in which the extreme vertices in distinct copies of $S(n-1,k)$ are connected as the complete graph K_{k+1}.

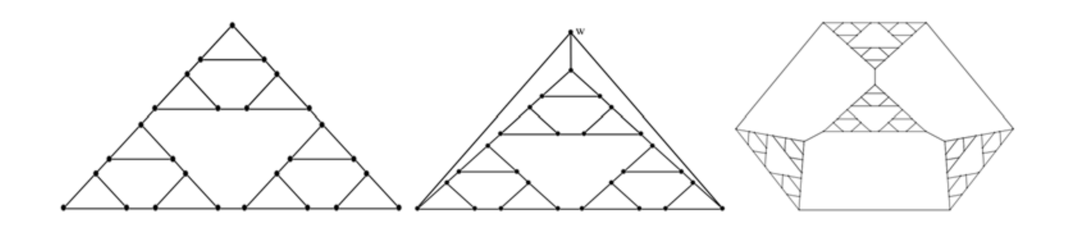


Figure 1: Graphs $S(3,3)$, $S^{+}(3,3)$, $S^{++}(4,3)$.

In [8], vertex, edge, total colorings of $S(n,k)$, $S^{+}(n,k)$, $S^{++}(n,k)$ are studied. In particular the authors give the following theorems:

Theorem 2. *For all integers* $n > 1$ *and* $k > 1$, *we have*

$$\chi'\big(S(n,k)\big) = k.$$

Theorem 3. *For all integers* $n > 1$ *and* $k > 1$, *we have*

$$\chi'\big(S^{+}(n,k)\big) = \begin{cases} k & \text{if } k \text{ is odd,} \\ k+1 & \text{if } k \text{ is even.} \end{cases}$$

Theorem 4. *For all integers* $n > 1$ *and* $k > 1$, *we have*

$$\chi'\left(S^{++}(n,k)\right) = k.$$

Worth to mentioning that these two families are the first known examples of "fractal" type of graphs whose crossing numbers can be determined.

In this paper, we study the palette indices of graphs $S(n,k)$, $S^{+}(n,k)$, $S^{++}(n,k)$. In particular, the palette index of $S(n,k)$ is determined when k is even, $k = 3$, or $n = 2$ and $k \equiv 3 \pmod 4$. Also a lower and an upper bound are given for $\check{s}\left(S(n,k)\right)$. The palette index of $S^{+}(n,k)$ is determined when k is odd, $k \equiv 0 \pmod 4$ and a lower and an upper bound are given when $k \equiv 2 \pmod 4$. At last, the palette index of $S^{++}(n,k)$ is determined.

Main Results

In this section we will examine the palette index of $S(n,k)$, $S^{+}(n,k)$, and $S^{++}(n,k)$ where $n, k > 1$.

The Sierpiński graphs $S(n,k)$ have two kinds of vertices in terms of vertex degree: extreme vertices with degree $k-1$ and the remaining with degree k. Thus, $\check{s}\left(S(n,k)\right) \geq 2$.

Proposition 2. *For every integer*$n > 1$ *and for every even integer* $k > 1$, *we have*

$$\check{s}\left(S(n,k)\right) = 2.$$

Proof. As $\check{s}\left(S(n,k)\right) \geq 2$, we just need a coloring with two palettes. Firstly, we color all k^{n-1} copies of K_k in $S(n,k)$ with the palette $\{1, \dots, k-1\}$. Then, we color all link edges of $S(n,k)$ with the color k. Thus, we gave a coloring of $S(n,k)$ with two palettes: $\{1, \dots, k-1\}$ and $\{1, \dots, k\}$.

Let us consider the case when k is odd.

Proposition 3. *For every odd integer $k > 1$ and every integer $n > 1$, we have*

$$\check{s}(S(n,k)) \geq 3.$$

Proof. In [9, Claim in Theorem 4.1], it was proved that for every integer $n \geq 1$ and every odd integer $k > 1$, $\chi'(S(n,k)) = k$, and the palettes of extreme vertices of $S(n,k)$ are pairwise different. If we want to color $S(n,k)$ with two palettes, then we must use a single palette for coloring vertices with degree k, that in turn means we use only k colors. So we will have k palettes for extreme vertices. Hence, $\check{s}(S(n,k)) \geq 3$.

Proposition 4. *For every odd integer $k > 1$ and every integer $n > 1$, we have*

$$\check{s}(S(n,k)) \leq \begin{cases} \check{s}(K_k) & \text{if } n = 2, \\ \check{s}(K_k) + 1 & \text{if } n > 2. \end{cases}$$

Proof. Let $k > 1$ be an odd integer. We can color K_k with three or four palettes, $P_s, P_1, \dots, P_m$ where $m = 2,3$, and there is one vertex s with unique palette P_s[6]. In order to color $S(n,k)$, we color all $S(1,k)$ in $S(n,k)$ as colored in [6] in a way that all extreme vertices of $S(n,k)$ have the palette P_s. Then we color all link edges with a new color c. Thus we have $\check{s}(K_k)$ palettes: P_s, $P_1 \cup \{c\}, \dots, P_m \cup \{c\}$ if $n = 2$ and one more palette $P_s \cup \{c\}$ if $n > 2$.

Corollary 1. *For every integer $k \geq 0$, we have*

$$\check{s}(S(2,4k+3)) = 3.$$

Proof. The result follows from Theorem 1, Proposition 3, and Proposition 4.

Theorem 5. *For every integer $n > 1$, we have $\check{s}(S(n,3)) = 3$.*

Proof. Proposition 3 yields $\check{s}(S(n,3)) \geq 3$. So we need a coloring of $S(n,3)$ with three palettes to prove the theorem.

In Figure 2, a coloring of $S(2,3)$ with three palettes is given.

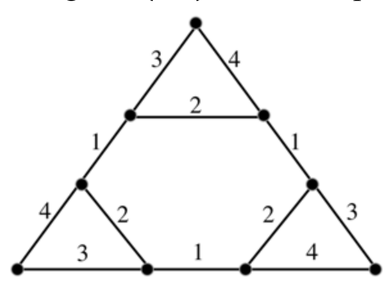


Figure 2: A coloring of S(2,3) with three palettes

In Figure 3, two colorings of $S(2,3)$ are given.

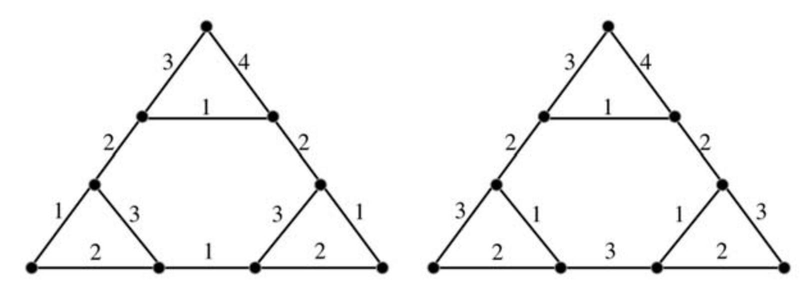


Figure 3: Group A.

Let us denote the group of sets of palettes from Figure 3 by A:

1. $\{\{1,2,3\},\{1,2,4\},\{1,2\},\{3,4\}\}$;
2. $\{\{1,2,3\},\{1,2,4\},\{2,3\},\{3,4\}\}$.

Next, let us consider the colorings of $S(n,3)$ in Figure 4.

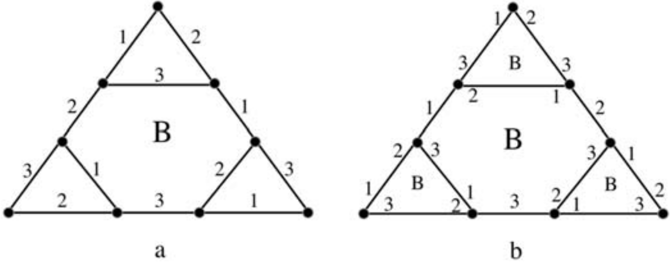


Figure 4: Group B.

Figure 4.a gives a coloring of $S(2,3)$. We denote this single element group of sets of palettes by B:

1. $\{\{1,2,3\},\{1,2\},\{1,3\},\{2,3\}\}$.

Figure 4.b shows how we can color S(n, 3) using palettes from group B.

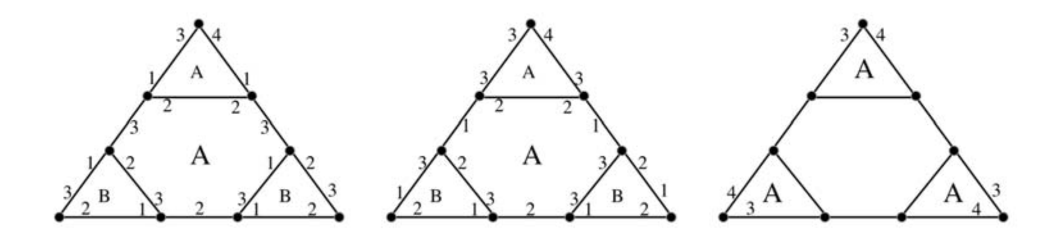


Figure 5: The coloring of $S(n,3)$ with three palettes

In Figure 5, we can see that we can color $S(n,3)$ using each set of palettes from group A by coloring one of $S_i(n,3)$ using a set of palettes from group A, two others using a set of palettes from group B, and link edges that connect these $S_i(n,3)$ with colors 1,2, or 3, where $i = 1,2,3$. We can also see that we can color $S(n,3)$ with three palettes by coloring all $S_i(n,3)$ using a set of palettes from group A, and link edges that connect these $S_i(n,3)$ with a color 1 or 3, where $i = 1,2,3$.

Theorem 6. *For all integers $n > 1$ and $k > 1$, we have*

$$\check{s}(S^+(n,k)) = \begin{cases} 1 & \text{if } k \equiv 1 \pmod 2, \\ 3 & \text{if } k \equiv 0 \pmod 4, \end{cases}$$

and

$$3 \leq \check{s}(S^+(n,k)) \leq 4 \text{ if } k \equiv 2 \pmod 4.$$

Proof. If k is odd then Proposition 1 and Theorem 3 yield $\check{s}(S^+(n,k)) = 1$. If k is even then $S^+(n,k)$ is of Class 2 therefor $\check{s}(S^+(n,k)) \geq 3$. When $n = 1$ then $S^+(n,k)$ is isomorphic to K_{k+1}. Let us

fix an edge coloring ϕ of K_{k+1} with $\check{s}(K_{k+1})$ palettes P_{ϕ}. In order to color $S^{+}(n,k)$, we consider each $S_i(n,k)$ in $S(n,k)$ of $S^{+}(n,k)$ as a vertex where $i=1,\dots,k$. These vertices and w form the graph K_{k+1} which we color with ϕ. The colors of edges incident to vertices of $S_i(n,k)$ but not in $S_i(n,k)$ form a palette from P_{ϕ}. For each $S_i(n,k)$, we consider that we have a vertex w which already have a palette from P_{ϕ} and we act the same way as for $S^{+}(n,k)$. In this way, we reduce the n to 1 and we color the graph $S^{+}(n,k)$ with all palettes of P_{ϕ}. Thus $\check{s}\left(S^{+}(n,k)\right) \leq \check{s}(K_{k+1})$. Since $\check{s}(K_n) = \equiv 3 \pmod 4$ then $\check{s}\left(S^{+}(n,k)\right) = \equiv 2 \pmod 4$.

Proposition 5. *For all integers* $n>1$ *and* $k>1$*, we have* $\check{s}\left(S^{++}(n,k)\right)=1$.

Proof. It follows from Proposition 1 and Theorem 4.

Acknowledgement. The author would like to thank P.A. Petrosyan for many useful comments and suggestions.

REFERENCES

1. *Avesani M., Bonisoli A., Mazzuoccolo G.*A family of multigraphs with large palette index. Ars Mathematica Contemporanea, 2019, 17. 115–124.
2. *Bonisoli A., Bonvicini S., Mazzuoccolo G***.** On the palette index of a graph: the case of trees. *Lecture Notes of SeminarioInterdisciplinare di Matematica*, 2017, 14. 49–55.
3. *Bonvicini S., Mazzuoccolo G.* Edge-colorings of 4-regular graphs with the minimum number of palettes. Graphs Combin, 2016, 32, 1293–1311.
4. Casselgren C.J., Petrosyan P.A. Some results on the palette index of graphs. Discrete Mathematics and Theoretical Computer Science, 2019, 21 #11.
5. *Hornak M., Hudak J.* On the palette index of complete bipartite graphs. DiscussionesMathematicae Graph Theory, 2018, 38. 463–476.
6. *Hornak M., Kalinowski R., Meszka M., Wozniak M.* Minimum number of palettes in edge colorings. Graphs Combin, 2014, 30. 619–626.

7. *Imran M., Sabeel-e-Hafi, Gao W., Farahani M.* On topological properties of Sierpiński networks. Chaos Soliton.Fract., 2017, 47.199–204.
8. *Jakovac M., Klavžar S.* Vertex-, edge-, and total-colorings of Sierpiński-like graphs. Discrete Mathematics, 2009, 309, #6. 1548–1556.
9. *Kaimanovich V.* Random walks on Sierpiński graphs: hyperbolicity and stochastic homogenization. Trends in Mathematics. Fractals in Graz 2001, 2003. 145–183.
10. *Klavžar S., Milutinović U.* Graphs $S(n, k)$ and a variant of the Tower of Hanoi problem. Czechoslovak Math. J., 1997, 47. 95–104.
11. *Klavžar S., Mohar B.* Crossing numbers of Sierpiński-like graphs. J. Graph Theory, 2005, 50.186–198.
12. Leven D., Galil Z. NP completeness of finding the chromatic index of regular graphs. Journal of Algorithms, 1983, 4, 35–44.
13. 13.*D.B.* West Introduction to Graph Theory. Prentice-Hall, 2001, (New Jersey).

О ПАЛИТРОВОМ ИНДЕКСЕ ГРАФОВ ВИДА СЕРПИНСКОГО

А.Б. Казарян

Департамент Информатики и прикладной математики
Ереванского Государственного Университета

АННОТАЦИЯ

В этой статье мы исследуем палитру индексов графов вида Серпинского. При правильной реберной раскраске графа G мы определяем палитру вершины $v \in V(G)$ как множество всех цветов, появляющихся на ребрах, смежных с v. Индекс палитры $\check{s}(G)$ графа G является минимальным числом различных палитр, встречающихся при всех правильных реберных раскрасках G. А именно: мы определяем индекс палитры графов Серпинского $S(n, k)$, где k четно, $k = 3$ или $n = 2$ и $k \equiv 3 \pmod 4$, и определяем верхнюю и нижнюю границы для оставшихся случаев. Кроме того, мы определяем индекс палитры графов $S^{+}(n, k)$, когда k нечетно или $k \equiv 0 \pmod 4$, и определяем верхнюю и нижнюю границы для остальных случаев. Более того, мы полностью определяем индекс палитры графов $S^{++}(n, k)$.

Ключевые слова: индекс палитры, графы Серпинского, графы вида Серпинского.

UDC 519.178

Поступила: 31.03.2021г.
Сдана на рецензию: 31.03.2021г.
Подписана к печати: 19.04.2021г.

COMPLEXITY RESULTS ON LOCALLY-BALANCED K-PARTITIONS OF GRAPHS

A. Gharibyan

Chair of Discrete Mathematics and Theoretical Informatics, YSU, Armenia

aramgharibyan@gmail.com

ABSTRACT

A *k-partition of a graph G* is a surjection $f: V(G) \to \{0,1,\dots,k-1\}$. A k-partition ($k \geq 2$) f of a graph G is a *locally-balanced with an open neighborhood* if for every $v \in V(G)$ and any $0 \leq i < j \leq k-1$, $||\{u \in N_G(v): f(u) = i\}| - |\{u \in N_G(v): f(u) = j\}|| \leq 1$. A k-partition ($k \geq 2$) f' of a graph G is a *locally-balanced with a closed neighborhood* if for every $v \in V(G)$ and any $0 \leq i < j \leq k-1$, $||\{u \in N_G[v]: f'(u) = i\}| - |\{u \in N_G[v]: f'(u) = j\}|| \leq 1$. In this paper we prove that the problem of the existence of locally-balanced k-partition with an open (closed) neighborhood is NP-complete for bipartite graphs.

Keywords: k-partition, locally-balanced k-partition, NP-completeness, bipartite graph.

Introduction

In this paper all graphs are finite, undirected, and have no loops or multiple edges. Let $V(G)$ and $E(G)$ denote the sets of vertices and edges of a graph G, respectively. The set of neighbors of a vertex v in G is denoted by $N_G(v)$. Let $N_G[v] = N_G(v) \cup \{v\}$. The degree of a vertex $v \in V(G)$ is

denoted by $d_G(v)$ and the maximum degree of vertices in G by $\Delta(G)$. The terms and concepts that we do not define can be found in [1, 2].

The concept of locally-balanced 2-partition of graphs was introduced by Balikyan and Kamalian [3] in 2005 and motivated by the problems concerning a distribution of influences of two different powers, which minimizes the probability of conflicts. The subjects of a modelling system may or may not have an ability of self-defense. In [4], 2-vertex-colorings of graphs were considered for which each vertex is adjacent to the same number of vertices of every color. In particular, Kratochvil [4] proved that the problem of the existence of such a coloring is NP-complete even for the $(2p, 2q)$-biregular $(p, q \geq 2)$ bipartite graphs. Moreover, he also showed that the problem of the existence of the aforementioned coloring for the $(2,2q)$-biregular $(q \geq 2)$ bipartite graphs can be solved in polynomial time. Gerber and Kobler [5,6] suggested to consider the problem of deciding if a given graph has a 2-partition with nonempty parts such that each vertex has at least as many neighbors in its part as in the other part. In [7], it was proved that the problem is NP-complete. In [3], Balikyan and Kamalian proved that the problem of existence of locally-balanced 2-partition with an open neighborhood of bipartite graphs with maximum degree 3 is NP-complete. In 2006, the similar result for locally-balanced 2-partitions with a closed neighborhood was also proved [8]. In [9], were shown that the problem of existence of locally-balanced 2-partition with an open neighborhood of a (3,8)-biregular bipartite graph is NP-complete.Gharibyan also proved [10] that the problem of existence of locally-balanced 2-partition with a closed neighborhood of bipartite graphs with maximum degree 3 is NP-complete.

In the present paper we study the complexity of the problem of the existence of locally-balanced k-partition with an open (closed) neighborhood of graphs.In particular, we prove that the problem of the existence of locally-balanced k-partition with an open (closed) neighborhood is NP-complete for bipartite graphs.

Main Results

In this section we introduce some terminology and notation. If φ is a k-partition of a graph G and $v \in V(G)$, then for any $i \in \{0,1,\dots,k-1\}$, we define $\#(v)_{\varphi}^{i}$ and $\#[v]_{\varphi}^{i}$ as follows:
$\#(v)_{\varphi}^{i} = |\{u \in N_G(v) : \varphi(u) = i\}|$ and $\#[v]_{\varphi}^{i} = |\{u \in N_G[v] : \varphi(u) = i\}|$.

If φ is a k-partition of a graph G and $v \in V(G)$, then for any i,j ($0 \leq i,j \leq k-1$), we define $\#(v)_{\varphi}^{i,j}$ and $\#[v]_{\varphi}^{i,j}$ as follows:

$$\#(v)_{\varphi}^{i,j} = \#(v)_{\varphi}^{i} - \#(v)_{\varphi}^{j} \text{ and } \#[v]_{\varphi}^{i,j} = \#[v]_{\varphi}^{i} - \#[v]_{\varphi}^{j}.$$

We denote by $V = \{x_1,\dots,x_n\}$ a finite set of variables. A literal is either a variable x or a negated variable $\overline{x}$. We denote by $L_V = \{x, \overline{x} : x \in V\}$ the set of literals. A *clause* is a set of literals, i.e., a subset of L_V, and a k-*clause* is one which contains exactly k distinct literals. A clause is *monotone* if all of its involved variables contain no negations.
We define a function $NAE_n : \{0,1\}^n \to \{0,1\}$ in the following way:

$$NAE_n(x_1,x_2,\dots,x_n) = \begin{cases} 0, & if\ x_1 = x_2 = \cdots = x_n, \\ 1, & otherwise. \end{cases}$$

If c is a monotone k-clause and $x_{i_1}, x_{i_2}, \dots, x_{i_k} \in c$, then define $NAE_k(c)$ as follows:

$$NAE_k(c) = NAE_k(x_{i_1}, x_{i_2}, \dots, x_{i_k})$$

Let us consider the following

Problem 1. (NAE-3-Sat-E4). *Instance: Given a set $V = \{x_1,\dots,x_n\}$ of variables and a collection $C = \{c_1,\dots,c_k\}$ of monotone 3-clauses over V such that every variable appears in exactly four clauses.*

Question: Is $f(x_1,\dots,x_n) = NAE_3(c_1) \& \cdots \& NAE_3(c_k)$ formula satisfiable?

In [11], it was proved the following result.

Theorem 1. *Problem 1 is NP-complete.*

We need some lemmas from [12].

Lemma 2. *φ is a locally-balanced k-partition with an open neighborhood of a graph G if and only if for every $v \in V(G)$ and any $0 \leq i \leq k-1$,*

$$\left\lfloor \frac{d_G(v)}{k} \right\rfloor \leq \#(v)_\varphi^i \leq \left\lceil \frac{d_G(v)}{k} \right\rceil.$$

Lemma 3. *φ is a locally-balanced k-partition with a closed neighborhood of a graph G if and only if for every $v \in V(G)$ and any $0 \leq i \leq k-1$,*

$$\left\lfloor \frac{d_G(v)+1}{k} \right\rfloor \leq \#[v]_\varphi^i \leq \left\lceil \frac{d_G(v)+1}{k} \right\rceil.$$

Problem 2. *Instance: Given a bipartite graph G and a positive integer $3 \leq r \leq |V(G)|$.*

Question: Does G has a locally-balanced r-partition with an open neighbourhood?

Theorem 4. *Problem 2 is NP-complete.*

Proof. It is easy to see that Problem 2 is in NP. For the proof of the NP-completeness, we show a reduction from Problem 1 to Problem 2. Let $\mathcal{I} = (V, C)$ be an instance of Problem 1. We must construct a bipartite graph G and a positive integer $3 \leq r \leq |V(G)|$, such that it has a locally-balanced r-partition with an open neighbourhood if and only if $f(x_1, \dots, x_n) = NAE_3(c_1) \& \cdots \& NAE_3(c_k)$ formula is satisfiable.

Let us construct a graph G in such a way:

$$V(G)=\left(\bigcup_{i=1}^{k-2} v_i\right)\cup\left(\bigcup_{i=1}^{k-2} y_i\right)\cup\left(\bigcup_{i=1}^{n} p_i\right)\cup\left(\bigcup_{i=1}^{k}\bigcup_{j=1}^{k} q_i^j\right)\cup\left(\bigcup_{i=1}^{n}\bigcup_{j=1}^{k} u_i^j\right),$$

$$E(G)=\{u_j^l v_i\colon 1\le i\le k-2,\ 1\le j\le n,\ 1\le l\le k\}$$
$$\cup\{q_j^l y_i\colon 1\le i\le k-2,\ 1\le j\le k,\ 1\le l\le k\}$$
$$\cup\{u_i^l p_i\colon 1\le i\le n,\ 1\le l\le k\}$$
$$\cup\{q_j^l p_i\colon 1\le i\le n,\ 1\le l\le k,\ 1\le j\le k,\ x_i\in c_j\}.$$

The graph G has $(n+k+2)k+n-4$ vertices. Let $r=k$. Clearly, $3\le r\le |V(G)|$. It is easy to see that the graph can be constructed from V and C in polynomial time. Suppose that $(\beta_1,\dots,\beta_n)$ is a true assignment of $f(x_1,\dots,x_n)$. We show that G has a locally-balanced r-partition with an open neighbourhood. Let us define a k-partition φ of G as follows: for every $w\in V(G)$, let

$$\varphi(\mathrm{w})=\begin{cases} i+1,\ if\ w=y_i, where\ 1\le i\le k-2,\\ i-1,\ if\ w=q_i^j, where\ 1\le\ i\le\ k, 1\le\ j\le\ k,\\ i-1,\ if\ w=u_i^j, where\ 1\le i\le k, 1\le j\le n,\\ \quad\beta_i,\ if\ w=p_i, where\ 1\le i\le n.\end{cases}$$

Let us show that φ is a locally-balanced k-partition with an open neighbourhood. We consider five cases.

Case 1: Let $v_l\in V(G)(1\le l\le k-2)$.
By the construction of G and the definition of φ, we obtain $\#(v_l)_\varphi^i=n(0\le i\le k-1)$, thus $\forall i,j\ 0\le i\le k-1,\ 0\le j\le k-1,\ i\ne j\#(v_l)_\varphi^{i,j}=0$.

Case 2: Let $y_l\in V(G)(1\le l\le k-2)$.
By the construction of G and the definition of φ, we have $\#(y_l)_\varphi^i=k(0\le i\le k-1)$, thus $\forall i,j\ 0\le i\le k-1,\ 0\le j\le k-1,\ i\ne j\#(y_l)_\varphi^{i,j}=0$.

Case 3: Let $u_l^t\in V(G)(1\le l\le n,\ 1\le t\le k)$.
By the construction of G and the definition of φ, we get $0\le\#(u_l^t)_\varphi^i\le 1(0\le i\le k-1)$, thus

$$\forall i,j\ 0\le i\le k-1,\ 0\le j\le k-1,\ i\ne j\left|\#(u_l^t)_\varphi^{i,j}\right|\le 1.$$

Case 4: Let $p_l \in V(G)(1 \le l \le n)$.
By the construction of G, the definition of φ and taking into account that $|\{p_l q_j^i : p_l q_j^i \in E(G),\ 1 \le i \le r, 1 \le j \le k\}| = 4k$ we have $\#(p_l)_\varphi^i = 5(0 \le i \le k-1)$, thus

$$\forall i,j\ 0 \le i \le k-1,\ 0 \le j \le k-1,\ i \ne j \#(p_l)_\varphi^{i,j} = 0.$$

Case 5: Let $q_l^t \in V(G)(1 \le l \le k,\ 1 \le t \le k)$.

Let $x_{s_1}, x_{s_2}, x_{s_3} \in c_l$, thus $NAE_3(\beta_{s_1}, \beta_{s_2}, \beta_{s_3}) = 1$. Form this, by the construction of G and the definition of φ we get $1 \le \#(q_l^t)_\varphi^i \le 2(0 \le i \le k-1)$, thus $\forall i,j\ 0 \le i \le k-1,\ 0 \le j \le k-1,\ i \ne j \left|\#(q_l^t)_\varphi^{i,j}\right| \le 1$.

Conversely, suppose that α is a locally-balanced k-partition with an open neighbourhood of G. Since $d_G(u_i^j) = k-1(1 \le i \le n,\ 1 \le j \le k)$ by Lemma 2 we have

$$\alpha(p_i) \ne \alpha(v_t)\ (1 \le t \le k-2),$$
$$\alpha(v_s) \ne \alpha(v_t)\ (1 \le t \le k-2,\ 1 \le s \le k-2).$$

From this we obtain

$$\left|\{\alpha(p_j):\ 1 \le j \le n\}\right| \le 2. (1)$$

Let us consider a vertex $q_l^t(1 \le l \le k,\ 1 \le t \le k)$. From Lemma 2 and taking into account that $d_G(q_l^t) = k+1$, we get

$$\exists p_{j'}, p_{j''} \in N_G(q_l^t)\ \alpha(p_{j'}) \ne \alpha(p_{j''}). (2)$$

From (1) and (2) we get $\left|\{\alpha(p_j):\ 1 \le j \le n\}\right| = 2$. Thus $i_1, i_2 \in \{\alpha(p_j):\ 1 \le j \le n\}$ and $i_1 \ne i_2$. Let us define an assignment of $f(x_1, \dots, x_n)$ as follows: $\forall 1 \le j \le n$

$$x_j = \begin{cases} 0,\ if\ \alpha(p_j) = i_1, \\ 1,\ if\ \alpha(p_j) = i_2. \end{cases}$$

Let $c_i \in C(1 \le i \le k)$ and $c_i = \{x_{j_1}, x_{j_2}, x_{j_3}\}$. From (2) we get $NAE_3(x_{j_1}, x_{j_2}, x_{j_3}) = 1$, which implies that $f(x_1, \dots, x_n)$ is satisfiable.

Problem 3. *Instance: A bipartite graph* G*and a positive integer* $3 \leq r \leq |V(G)|$.

Question: Does G *has a locally-balanced* r*-partition with a closed neighbourhood?*

Theorem 5. *Problem* 3 *is* NP*-complete.*

Proof. It is easy to see that Problem 3 is in NP. For the proof of the NP-completeness, we show a reduction from Problem 1 to Problem 3. Let $\mathcal{I} = (V, C)$ be an instance of Problem 1. We must construct a bipartite graph G and a positive integer $3 \leq r \leq |V(G)|$, such that it has a locally-balanced r-partition with a closed neighbourhood if and only if $f(x_1, \dots, x_n) = NAE_3(c_1) \& \cdots \& NAE_3(c_k)$ formula is satisfiable.

Let us construct a graph G in such a way:

$$V(G) = \left(\bigcup_{i=1}^{k}\bigcup_{j=1}^{2} z_i^j\right) \cup \left(\bigcup_{i=1}^{k-2} y_i\right) \cup \left(\bigcup_{i=1}^{k}\bigcup_{j=1}^{k-2} q_i^j\right) \cup \left(\bigcup_{i=1}^{n} p_i\right) \cup \left(\bigcup_{i=1}^{n}\bigcup_{j=1}^{7} u_i^j\right)$$
$$\cup \left(\bigcup_{i=1}^{n}\bigcup_{j=5}^{7}\bigcup_{l=1}^{k-1} s_i^j(l)\right) \cup \left(\bigcup_{i=1}^{k-2} v_i\right) \cup \left(\bigcup_{i=1}^{n}\bigcup_{j=1}^{k} b_i^j\right) \cup \left(\bigcup_{i=1}^{n} a_i\right),$$

$$E(G) = \{z_j^l y_i\colon\ 1 \leq j \leq k,\ 1 \leq l \leq 2,\ 1 \leq i \leq k-2\}$$
$$\cup \{q_j^l y_i\colon\ 1 \leq j \leq k,\ 1 \leq l \leq k-2,\ 1 \leq i \leq k-2\}$$
$$\cup \{q_j^l p_i\colon\ 1 \leq j \leq k,\ 1 \leq l \leq k-2,\ 1 \leq i \leq n,\ x_i \in c_j\}$$
$$\cup \{u_j^l p_j\colon\ 1 \leq j \leq n,\ 1 \leq l \leq 7\}$$
$$\cup \{u_j^l s_j^l(i)\colon\ 1 \leq j \leq n,\ 5 \leq l \leq 7,\ 1 \leq i \leq k-1\}$$
$$\cup \{u_j^1 v_i\colon\ 1 \leq j \leq n,\ 1 \leq i \leq k-2\}$$
$$\cup \{a_j v_i\colon\ 1 \leq j \leq n,\ 1 \leq i \leq k-2\}$$
$$\cup \{b_j^l v_i\colon\ 1 \leq j \leq n,\ 3 \leq l \leq k,\ 1 \leq i \leq k-2\}$$
$$\cup \{b_j^l b_j^i\colon\ 1 \leq j \leq n,\ 1 \leq l \leq 2,\ 3 \leq i \leq k\}.$$

The graph G has $n(6 + 4k) + k(2 + k) - 4$ vertices. Let $r = k$. Clearly, $3 \leq r \leq |V(G)|$. It is easy to see that the graph can be constructed

from V and C in polynomial time. Suppose that $(\beta_1, \dots, \beta_n)$ is a true assignment of $f(x_1, \dots, x_n)$. We show that G has a locally-balanced r-partition with a closed neighbourhood. Let us define a k-partition φ of G as follows: for every $w \in V(G)$, let

$$\varphi(w) = \begin{cases} j-1, \; if \; w = z_i^j, where \; 1 \le i \le k, 1 \le j \le 2, \\ i+1, if \; w = y_i, where \; 1 \le i \le k-2, \\ j+1, \; if \; w = q_i^j, where \; 1 \le i \le k, 1 \le j \le k-2, \\ \beta_i, \; if \; w = p_i, where \; 1 \le i \le n, \\ \beta_i, \; if \; w = u_i^j, where \; 1 \le i \le n, 5 \le j \le 7, \\ 1-\beta_i, \; if \; w = s_i^j(1), where \; 1 \le i \le n, 5 \le j \le 7, \\ l, \; if \; w = s_i^j(l), where \; 1 \le i \le n, 5 \le j \le 7, 2 \le l \le k-1, \\ 1-\beta_i, \; if \; w = u_i^j, where \; 1 \le i \le n, 1 \le j \le 4, \\ i+1, \; if \; w = v_i, where \; 1 \le \; i \le \; k-2, \\ \beta_i, \; if \; w = a_i, where \; 1 \le i \le n, \\ j-1, \; if \; w = b_i^j, where \; 1 \le i \le n, 1 \le j \le k. \end{cases}$$

Let us show that φ is a locally-balanced k-partition with a closed neighbourhood. We consider 12 cases.

Case 1: Let $z_l^t \in V(G)(1 \le l \le k, \; 1 \le t \le 2)$.
By the construction of G and the definition of φ, we obtain $0 \le \#[z_l^t]_\varphi^i \le 1 (0 \le i \le k-1)$, which implies $\forall i,j \; 0 \le i \le k-1, \; 0 \le j \le k-1, \; i \ne j \left|\#[z_l^t]_\varphi^{i,j}\right| \le 1$.

Case 2: Let $y_l \in V(G)(1 \le l \le k-2)$.
By the construction of G and the definition of φ, we obtain $k \le \#[v_l]_\varphi^i \le k+1 (0 \le i \le k-1)$, thus $\forall i,j \; 0 \le i \le k-1, \; 0 \le j \le k-1, \; i \ne j \left|\#[y_l]_\varphi^{i,j}\right| \le 1$.

Case 3: $q_l^t \in V(G)(1 \le l \le k, \; 1 \le t \le k-2)$.

Let $x_{s_1}, x_{s_2}, x_{s_3} \in c_l$, thus $NAE_3(\beta_{s_1}, \beta_{s_2}, \beta_{s_3}) = 1$. Form this, by the construction of G and by the definition of φ we get $1 \leq \#[q_l^t]_\varphi^i \leq 2(0 \leq i \leq k-1)$, thus $\forall i,j\ 0 \leq i \leq k-1,\ 0 \leq j \leq k-1,\ i \neq j \left|\#[q_l^t]_\varphi^{i,j}\right| \leq 1$.

Case 4: Let $p_l \in V(G)(1 \leq l \leq n)$.
By the construction of G and the definition of φ, we have $\#[p_l]_\varphi^i = 4(0 \leq i \leq r-1)$, thus $\forall i,j\ 0 \leq i \leq r-1,\ 0 \leq j \leq r-1,\ i \neq j \#[p_l]_\varphi^{i,j} = 0$.

Case 5: Let $u_l^t \in V(G)(1 \leq l \leq n,\ 2 \leq t \leq 4)$.
By the construction of G and the definition of φ, we obtain $0 \leq \#[u_l^t]_\varphi^i \leq 1(0 \leq i \leq k-1)$, which implies

$$\forall i,j\ 0 \leq i \leq k-1,\ 0 \leq j \leq k-1,\ i \neq j \left|\#[u_l^t]_\varphi^{i,j}\right| \leq 1.$$

Case 6: Let $u_l^t \in V(G)(1 \leq l \leq n,\ 5 \leq t \leq 7)$.
By the construction of G and the definition of φ, we obtain $1 \leq \#[u_l^t]_\varphi^i \leq 2(0 \leq i \leq k-1)$, which implies

$$\forall i,j\ 0 \leq i \leq k-1,\ 0 \leq j \leq k-1,\ i \neq j \left|\#[u_l^t]_\varphi^{i,j}\right| \leq 1.$$

Case 7: Let $u_l^1 \in V(G)(1 \leq l \leq n)$.
By the construction of G and the definition of φ, we obtain $\#[u_l^1]_\varphi^i = 1(0 \leq i \leq k-1)$, which implies

$$\forall i,j\ 0 \leq i \leq k-1,\ 0 \leq j \leq k-1,\ i \neq j \#[u_l^1]_\varphi^{i,j} = 0.$$

Case 8: Let $s_l^t(h) \in V(G)(1 \leq l \leq n,\ 5 \leq t \leq 7,\ 1 \leq h \leq k-1)$.
By the construction of G and the definition of φ, we obtain $0 \leq \#[s_l^t(h)]_\varphi^i \leq 1(0 \leq i \leq k-1)$, which implies

$$\forall i,j\ 0 \leq i \leq k-1,\ 0 \leq j \leq k-1,\ i \neq j \left|\#[s_l^t(h)]_\varphi^{i,j}\right| \leq 1.$$

Case 9: Let $v_l \in V(G)(1 \leq l \leq k-2)$.
By the construction of G and the definition of φ, we obtain $n \leq \#[v_l]_\varphi^i \leq n+1(0 \leq i \leq k-1)$, which implies

$$\forall i,j\ 0 \leq i \leq k-1,\ 0 \leq j \leq k-1,\ i \neq j \left|\#[v_l]_\varphi^{i,j}\right| \leq 1.$$

Case 10: Let $a_l \in V(G)(1 \leq l \leq n)$.
By the construction of G and the definition of φ, we obtain $0 \leq \#[a_l]_\varphi^i \leq 1(0 \leq i \leq k-1)$, which implies

$$\forall i,j\ 0 \le i \le k-1,\ 0 \le j \le k-1,\ i \ne j \left|\#[a_l]_\varphi^{i,j}\right| \le 1.$$

Case 11: Let $b_l^t \in V(G)(1 \le l \le n,\ 3 \le t \le k)$.

By the construction of G and the definition of φ, we obtain $1 \le \#[b_l^t]_\varphi^i \le 2(0 \le i \le k-1)$, which implies

$$\forall i,j\ 0 \le i \le k-1,\ 0 \le j \le k-1,\ i \ne j \left|\#[b_l^t]_\varphi^{i,j}\right| \le 1.$$

Case 12: Let $b_l^t \in V(G)(1 \le l \le n,\ 1 \le t \le 2)$.

By the construction of G and the definition of φ, we obtain $0 \le \#[b_l^t]_\varphi^i \le 1(0 \le i \le k-1)$, which implies

$$\forall i,j\ 0 \le i \le k-1,\ 0 \le j \le k-1,\ i \ne j \left|\#[b_l^t]_\varphi^{i,j}\right| \le 1.$$

Conversely, suppose that α is a locally-balanced k-partition with a closed neighbourhood of G. Since $d_G(u_i^1) = k-1(1 \le i \le n)$ by Lemma 3 we have

$$\alpha(p_i) \ne \alpha(v_t)\ (1 \le t \le k-2),$$
$$\alpha(v_s) \ne \alpha(v_t)\ (1 \le t \le k-2,\ 1 \le s \le k-2).$$

Which implies

$$\left|\{\alpha(p_j):\ 1 \le j \le n\}\right| \le 2. (3)$$

Let us consider a vertex $q_l^t(1 \le l \le k,\ 1 \le t \le k-2)$. From Lemma 3 and taking into account that $d_G(q_l^t) = k+1$, we get

$$\exists p_{j'}, p_{j''} \in N_G(q_l^t)\ \alpha(p_{j'}) \ne \alpha(p_{j''}). (4)$$

From (3) and (4) we get $\left|\{\alpha(p_j):\ 1 \le j \le n\}\right| = 2$. Thus $i_1, i_2 \in \{\alpha(p_j):\ 1 \le j \le n\}$ and $i_1 \ne i_2$. Let us define an assignment of $f(x_1, \ldots, x_n)$ as follows: $\forall 1 \le j \le n$

$$x_j = \begin{cases} 0,\ if\ \alpha(p_j) = i_1, \\ 1,\ if\ \alpha(p_j) = i_2. \end{cases}$$

Let $c_i \in C(1 \le i \le k)$ and $c_i = \{x_{j_1}, x_{j_2}, x_{j_3}\}$. From (4) we get $NAE_3(x_{j_1}, x_{j_2}, x_{j_3}) = 1$, which implies that $f(x_1, \ldots, x_n)$ is satisfiable.

REFERENCES

1. *Chartrand G., Zhang P.* Chromatic Graph Theory, Discrete Mathematics and Its Applications. CRC Press (2009).
2. *West D.B*. Introduction to Graph Theory. N.J.,Prentice-Hall (2001).
3. *Balikyan S.V., Kamalian R.R.* On NP-Completeness of the Problem of Existence of Locally-balanced 2-partition for Bipartite Graphs G with $\Delta(G) = 3$. *Doklady NAN RA* 105: 1 (2005), 21–27.
4. *Kratochvil J.* Complexity of Hypergraph Coloring and Seidel's Switching. Graph Theoretic Concepts in Computer Science, 29th International Workshop, WG 2003, Elspeet, The Netherlands, Revised Papers, 2880 (2003), 297–308.
5. *Gerber M., Kobler D.*Partitioning a graph to satisfy all vertices, Technical report, Swiss Federal Institute of Technology, Lausanne, 1998.
6. *Gerber M., Kobler D.* Algorithmic approach to the satisfactory graph partitioning problem, European J. Oper. Res. 125 (2000) 283–291.
7. *Bazgan C., TuzaZs., Vanderpooten D.* The satisfactory partition problem, Discrete Applied Mathematics 154 (2006) 1236–1245.
8. *Balikyan S.V., Kamalian R.R.* On NP-completeness of the Problem of Existence of Locally-balanced 2-partition for Bipartite Graphs G with $\Delta(G) = 4$ Under the Extended Definition of the Neighbourhood of a Vertex. *Doklady NAN RA*, 106: 3 (2006), 218–226.
9. *Gharibyan A.H., Petrosyan P.A.* On locally-balanced 2-partitions ofbipartite graphs, Proceedingsof the Yerevan State University, Physicaland Mathematical Sciences 54:3 (2020), 137–145.
10. *Garibyan. A.H.* Two NP-Complete problems on locally-balanced 2-partitions of graphs,ВестникРоссийско-Армянскогоуниверситета,Физико-Математические и естественныенауки2 (2020), 5-21.
11. *Darmann A., Döcker J.* On a simple hard variant of Not-All-Equal 3-Sat. Theoretical Computer Science, 815 (2020), 147–152.
12. *Gharibyan A.H., Petrosyan P.A.* Locally-balanced k-partitions of graphs, Proceedings of the Yerevan State University, Physical and Mathematical Sciences 55:1 (2021), (Submitted).

О СЛОЖНОСТИ ЛОКАЛЬНО-СБАЛАНСИРОВАННЫХ К-РАЗБИЕНИЙ ГРАФОВ

А.Г. Гарибян

Кафедра Дискретной математики и теорретической информатики ЕГУ, Армения

АННОТАЦИЯ

В данной статье речь идет о локально сбалансированных k-разбиением графа G, которые называются сюръекцией $f: V(G) \to \{0,1,\dots,k-1\}$. k-разбиение $(k \geq 2) f$ графа G является локально-сбалансированным с открытой окрестностью, если для любой вершины $v \in V(G)$ и любых $0 \leq i < j \leq k-1$, $||\{u \in N_G(v), f(u) = i\}| - |\{u \in N_G(v), f(u) = j\}|| \leq 1$. k-разбиение $(k \geq 2) f$ графа G является локально-сбалансированным с закрытой окрестностью, если для любой вершины $v \in V(G)$ и любых $0 \leq i < j \leq k-1$, $||\{u \in N_G[v], f(u) = i\}| - |\{u \in N_G[v], f(u) = j\}|| \leq 1$, где $N_G[v] = N_G(v) \cup \{v\}$. В настоящей работе доказано, что задача существования локально-сбалансированных k-разбиений с открытой (закрытой) окрестностью NP полна даже в случае двудольных графов.

Ключевые слова: локально-сбалансированное k-разбиение, NP-полнота, двудольный граф.

УДК 517.518.23 Поступила: 12.04.2021г.
Сдана на рецензию: 13.04.2021г.
Подписана к печати: 19.04.2021г.

О СЛЕДАХ ФУНКЦИЙ ИЗ МУЛЬТИАНИЗОТРОПНЫХ ПРОСТРАНСТВ СОБОЛЕВА

М.А. Хачатурян, А.Р. Акопян

Российско-Армянский Университет

khmikayel@gmail.com, alexander.hakobyan@gmail.com

АННОТАЦИЯ

В данной научной статье получены прямые и обратные теоремы вложения разных измерений (теоремы о следах) для функций из мультианизотропного пространства Соболева $\mathrm{W}_2^{\mathfrak{N}}(\mathbb{R}^3)$ в случае одного класса вполне правильных многогранников $\mathfrak{N}$.

Ключевые слова: вполне правильный многогранник, мультианизотропное пространство Соболева, след функции.

Введение

Теория анизотропных пространств Соболева $W_p^{(l_1,l_2,\dots,l_n)}(\mathbb{R}^n)$ в случае $p = 2$ полностью разработано Л.Н. Слободецким [2,5]. Им же также было введено обобщение таких пространств на дробные порядки, с помощью чего были получены прямые и обратные теоремы вложения разных измерений для этих пространств. При изучении некоторого класса гипоэлиптических уравнений, введенным Л. Херманде-

ром, возникла необходимость изучения мультианизотропных функциональных пространств Соболева $W_p^{\mathfrak{N}}(\mathbb{R}^n)$, определяемых с помощью вполне правильного многогранника $\mathfrak{N}$ [6, 12], и являющимися обобщением анизотропных пространств Соболева. Впервые пространства такого типа изучались в работах С.М. Никольского и В.П. Михайлова. Исследования этих пространств было продолжено в работах Г.Г. Казаряна. Развитию теории мультианозотропных пространств посвящен цикл работ Г.А. Карапетяна [1, 7–11].

В данной работе для одного частного типа мультианизотропного пространства $W_2^{\mathfrak{N}}(\mathbb{R}^3)$, где трехмерный вполне правильный многогранник $\mathfrak{N}$ и из себя представляет пирамиду с вершиной $(0,0,l_3)$, и с основанием $\mathfrak{N}_0$ был изучен след функции $f \in W_2^{\mathfrak{N}}(\mathbb{R}^3)$ и ее обобщенных производных $D_{x_3}^s f (0 \le s < l_3)$ на гиперплоскости $x_3 = a$. Оказалось, что след принадлежит мультианизотропному пространству $W_2^{\mathfrak{N}_0{}'}(\mathbb{R}^2)$, где многогранник $\mathfrak{N}_0'$ подобен многограннику $\mathfrak{N}$ и немного меньше его, причем коэффициент подобия зависит от s и l_3. Все полученные результаты в работе имеют обратимый характер.

1. Определения, обозначения и вспомогательные результаты

Пусть $\mathbb{R}^n - n$-мерное Евклидово пространство точек $x = (x_1, x_2, \dots, x_n)$ с нормой $|x| := (\sum_{k=1}^n x_k^2)^{\frac{1}{2}}$, $\mathbb{Q}_+$ и $\mathbb{Z}_+$ – множества, соответственно, рациональных и целых неотрицательных чисел, $\mathbb{Q}_+^n$ – множество n-мерных векторов с компонентами из $\mathbb{Q}_+$, $\mathbb{Z}_+^n$ – множество n-мерных мультииндексов $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$, где $\alpha_j \in \mathbb{Z}_+$, $j = 1,2,\dots,n$. Для $x, \xi \in \mathbb{R}^n$, $t \in \mathbb{Z}_+$ и $\alpha \in \mathbb{Z}_+^n$ обозначим $|\alpha| := \alpha_1 + \dots + \alpha_n$, $x\xi = \sum_{k=1}^n x_k \xi_k$, $\xi^t := (\xi_1^t, \xi_2^t, \dots, \xi_n^t)$, $\xi^\alpha := \xi_1^{\alpha_1} \xi_2^{\alpha_2} \dots \xi_n^{\alpha_n}$, $D^\alpha := D_{x_1}^{\alpha_1} D_{x_2}^{\alpha_2} \dots D_{x_n}^{\alpha_n}$, понимаемая как *обобщенная производная,* по С.Л. Соболеву, порядка α.

Через $[a]$ и a обозначим, соответственно, целые и дробные части числа $a \in \mathbb{R}$, так что $a = [a] + a$. При $r \in \mathbb{Q}_+^n$ обозначим $[r] := ([r_1], [r_2], \dots, [r_n])$.

Для произвольной функции $f(x)$ и числа $t \in \mathbb{R}$ обозначим:

$$\Delta_j(t) f(x) := f\left(x + t e^{(j)}\right) - f(x),$$

где $e^{(j)} = (0, \dots, 0, 1, 0, \dots, 0)$ – j-ый координатный вектор в пространстве $\mathbb{R}^n$.

Через $S(\mathbb{R}^n)$ обозначим класс Шварца быстро убывающих на бесконечности функций. Через $F[\varphi](\xi)$ обозначим преобразование Фурье функции $\varphi \in S(\mathbb{R}^n)$, определяемое по формуле

$$F[\varphi](\xi) := \frac{1}{(2\pi)^{\frac{n}{2}}} \int_{\mathbb{R}^n} e^{-ix\xi}\, \varphi(x) dx.$$

Аналогично, обратное преобразование Фурье функции $\psi \in S(\mathbb{R}^n)$ обозначим через $F^{-1}[\psi](x)$, определяемое по формуле

$$F^{-1}[\psi](x) = \frac{1}{(2\pi)^{\frac{n}{2}}} \int_{\mathbb{R}^n} e^{ix\xi}\, \psi(\xi) d\xi.$$

Пусть $1 \le k \le n$, $x = (x', x'')$, $\xi = (\xi', \xi'')$, где $x' = (x_1, x_2, \dots, x_k) \in \mathbb{R}^k$, и $x'' = (x_{k+1}, \dots, x_n) \in \mathbb{R}^{n-k}$. Частичное преобразование Фурье-функции $\varphi(x) \in S(\mathbb{R}^n)$ относительно группы переменных x' и частичное обратное преобразование Фурье-функции $\psi(\xi) \in S(\mathbb{R}^n)$ относительно группы переменых ξ' будем обозначать через $F_{x'}[\varphi](\xi', x'')$ и $F_{\xi'}^{-1}[\psi](x', \xi'')$, соответственно.

Для заданного набора векторов из $\mathbb{Q}_+^n$ через $\mathfrak{N}$ обозначим наименьший выпуклый многогранник, содержащий все точки этого набора. Многогранник $\mathfrak{N} \subset \mathbb{R}_+^n$ называется *вполне правильным*, если он имеет вершину в начале координат и отличные от начала координат вершины на каждой оси координат, а внешние нормали всех $(n-1)$-мерных некоординатных граней имеют положительные компоненты. Обозначим через $r^{(k)} (k = 1, 2, \dots, M)$ вершины многогранника $\mathfrak{N}$, отличные от нуля, которые будем называть *главными вершинами*. Для произвольного $m \in \mathbb{Q}_+$ через $m\mathfrak{N}$ будем обозначать многогранник с

главными вершинами $mr^{(k)}(k = 1,2, \ldots, M)$. Для заданного вполне правильного многогранника $\mathfrak{N} \subset \mathbb{R}^n$ обозначим $P_{\mathfrak{N}}(\xi) := 1 + \sum_{k=1}^{M} (\xi^2)^{r^{(k)}}$.

Для заданного вполне правильного многогранника $\mathfrak{N}$, главными вершинами которого являются мультииндексы $\alpha^{(k)} \in \mathbb{Z}_n^+ (k = 1,2, \ldots, M)$, через $W_2^{\mathfrak{N}}(\mathbb{R}^n)$ обозначим *мультианизотропное пространство Соболева*, определяемое следующим образом:

$$W_2^{\mathfrak{N}}(\mathbb{R}^n) = f\colon f \in L_2(\mathbb{R}^n)\ \&\ D^{\alpha^{(k)}} f \in L_2(\mathbb{R}^n)\ \forall k = 1,2, \ldots, M,$$

наделенное нормой

$$\| f \|_{W_2^{\mathfrak{N}}(\mathbb{R}^n)} = \| f \|_{L_2(\mathbb{R}^n)} + \sum_{k=1}^{M} \| D^{\alpha^{(k)}} f \|_{L_2(\mathbb{R}^n)}. \qquad (1.1)$$

Норма (1.1) превращает $W_2^{\mathfrak{N}}(\mathbb{R}^n)$ в Банахово пространство.

Используя равенство Парсеваля, легко доказать следующие два утверждения.

Лемма 1.1. Пусть $\mathfrak{N}$ – вполне правильный многогранник, вершины которого являются мультииндексами. Тогда

$$W_2^{\mathfrak{N}}(\mathbb{R}^n) = u\colon u \in L_2(\mathbb{R}^n)\ \&\ \sqrt{P_{\mathfrak{N}}(\xi)} F[u](\xi) \in L_2(\mathbb{R}^n) \qquad (1.2)$$

и исходная норма в пространстве $W_2^{\mathfrak{N}}(\mathbb{R}^n)$ эквивалентна выражению

$$\left(\int_{\mathbb{R}^n} P_{\mathfrak{N}}(\xi) |F[u](\xi)|^2 d\xi \right)^{\frac{1}{2}} \qquad (1.3)$$

Лемма 1.2. Пусть $\mathfrak{N}$ есть вполне правильный многогранник, вершины которого являются мультииндексами, $u \in W_2^{\mathfrak{N}}(\mathbb{R}^n)$. Тогда

для любого мультииндекса $\beta \in \mathfrak{N}$ функция u имеет обобщенную производную $D^{\beta}u$, которая принадлежит пространству $L_2(\mathbb{R}^n)$.

Следующее утверждение непосредственно следует из теоремы 2.1 данной работы [12].

Лемма 1.3. Для любого вполне правильного многогранника $\mathfrak{N}$, вершины которого являются мультииндексами, множество $C_0^{\infty}(\mathbb{R}^n)$ плотно в $W_2^{\mathfrak{N}}(\mathbb{R}^n)$.

2. Пространство $W_2^{\mathfrak{N}}(\mathbb{R}^2)$ в случае многогранника с нецелыми вершинами

Пусть $\mathfrak{N}$ – вполне правильный многогранник с вершинами из $\mathbb{Q}_+^2$.

Определение 2.1. Скажем, что функция $f \in W_2^{\mathfrak{N}}(\mathbb{R}^2)$, если $f \in L_2(\mathbb{R}^n)$, $D^{[r^{(k)}]}f \in L_2(\mathbb{R}^2)$ $(k = 1, \dots, M)$, и для любого $r^{(k)}$ – такого, что $[r^{(k)}] \neq r^{(k)}$, конечен интеграл

$$I\left(r^{(k)}, f\right) = \\ = \begin{cases} \int\limits_{\mathbb{R}^3} \frac{\left|\Delta_1(z) D^{[r^{(k)}]} f(x)\right|^2}{|z|^{1+2r_1^{(k)}}} dx_1 dx_2 dz, \text{ если } r_1^{(k)} > 0,\ r_2^{(k)} = 0 \\ \int\limits_{\mathbb{R}^3} \frac{\left|\Delta_2(z) D^{[r^{(k)}]} f(x)\right|^2}{|z|^{1+2r_2^{(k)}}} dx_1 dx_2 dz, \text{ если } r_1^{(k)} = 0,\ r_2^{(k)} > 0 \\ \int\limits_{\mathbb{R}^4} \frac{\left|\Delta_1(z_1)\Delta_2(z_2) D^{[r^{(k)}]} f(x)\right|^2}{|z_1|^{1+2r_1^{(k)}} |z_2|^{1+2r_2^{(k)}}} dx_1 dx_2 dz_1 dz_2, \text{ если } r_1^{(k)} > 0,\ r_2^{(k)} > 0 \end{cases}. \tag{2.1}$$

Обозначим

$$N\left(r^{(k)}, f\right) = \begin{cases} I\left(r^{(k)}, f\right), \text{ если } \left[r^{(k)}\right] \neq r^k \\ \| D^{r^{(k)}} f \|_{L_2(\mathbb{R}^2)}^2, \text{ если } \left[r^{(k)}\right] = r^{(k)} \end{cases}. \tag{2.2}$$

Норма в пространстве $W_2^{\mathfrak{N}}(\mathbb{R}^2)$ вводится следующим образом:

$$\| f \|_{W_2^{\mathfrak{N}}(\mathbb{R}^2)} = \| f \|_{L_2(\mathbb{R}^n)} + \sum_{k=1}^{M} N\left(r^{(k)}, f\right)^{\frac{1}{2}}. \tag{2.3}$$

Лемма 2.1. Пусть $\mathfrak{N}$ – вполне правильный многогранник, вершины которого принадлежат $\mathbb{Q}_+^2$. Тогда

$$\mathrm{W}_2^{\mathfrak{N}}(\mathbb{R}^n) = \mathrm{u}: \mathrm{u} \in \mathrm{L}_2(\mathbb{R}^n) \,\&\, \sqrt{\mathrm{P}_{\mathfrak{N}}(\xi)}\mathrm{F}[\mathrm{u}](\xi) \in \mathrm{L}_2(\mathbb{R}^n), \tag{2.4}$$

и исходная норма в пространстве $\mathrm{W}_2^{\mathfrak{N}}(\mathbb{R}^n)$ эквивалентна выражению

$$\left(\int_{\mathbb{R}^n} \mathrm{P}_{\mathfrak{N}}(\xi)|\mathrm{F}[\mathrm{u}](\xi)|^2 d\xi\right)^{\frac{1}{2}}. \tag{2.5}$$

Доказательство. Пусть функция $u \in W_2^{\mathfrak{N}}(\mathbb{R}^n)$. Покажем, что $\sqrt{P_{\mathfrak{N}}(\xi)}F[u] \in L_2(\mathbb{R}^n)$. В силу определения $P_{\mathfrak{N}}(\xi)$ достаточно показать, что

$$\int_{\mathbb{R}^n} (\xi^2)^{r^{(k)}} |F[u](\xi)|^2 d\xi < \infty, \text{ где } k = 1, \dots, M. \tag{2.6}$$

В случае $\left[r^{(k)}\right] = r^{(k)}$ имеем, что $D^{r^{(k)}}u \in L_2(\mathbb{R}^2)$, следовательно, согласно свойствам преобразования Фурье,

$$(-i\xi)^{r^{(k)}}F[u](\xi) = F\left[D^{r^{(k)}}u\right](\xi) \in L_2(\mathbb{R}^2)$$

из чего следует конечность (2.6).

Пусть $\left[r^{(k)}\right] \neq r^{(k)}$. Приведем доказательство в случае $r_1^{(k)} > 0$, $r_2^{(k)} > 0$, так как в остальных случаях утверждение доказывается аналогичным образом. Рассмотрим интеграл

$$J_{r^{(k)}}(\xi_1, \xi_2) = \int_{\mathbb{R}^2} \frac{\left|e^{i\xi_1 z_1} - 1\right|^2 \left|e^{i\xi_2 z_2} - 1\right|^2}{|z_1|^{1+2r_1^{(k)}} |z_2|^{1+2r_2^{(k)}}} dz_1 dz_2 =$$

$$= \int\limits_{\mathbb{R}^2} \frac{\sin^2\left(\frac{\xi_1 z_1}{2}\right)}{|z_1|^{1+2r_1^{(k)}}} \cdot \frac{\sin^2\left(\frac{\xi_2 z_2}{2}\right)}{|z_2|^{1+2r_2^{(k)}}} dz_1 dz_2.$$

Так как очевидно, что интеграл $\int_{-\infty}^{+\infty} \frac{\sin^2(z)}{|z|^{1+2r}} dz$ сходится для $r \in (0,1)$, то учитывая, что в рассматриваемом случае $r_1^{(k)}, r_2^{(k)} \in (0,1)$, получим, что $J_{r^{(k)}}(\xi_1, \xi_2)$ конечен при любых значениях ξ_1 и ξ_2. Производя замену переменных $\tau_1 = |\xi_1| z_1$, $\tau_2 = |\xi_2| z_2$, получим, что

$$J_{r^{(k)}}(\xi_1, \xi_2) = (\xi^2)^{r^{(k)}} J_{r^{(k)}}(1,1). \tag{2.7}$$

Из (2.6) в силу (2.7) имеем, что

$$\int\limits_{\mathbb{R}^2} (\xi^2)^{r^{(k)}} |F[u](\xi)|^2 d\xi_1 d\xi_2 = \int\limits_{\mathbb{R}^2} (\xi^2)^{r^{(k)}} (\xi^2)^{[r^{(k)}]} |F[u](\xi)|^2 d\xi_1 d\xi_2 =$$

$$= \frac{1}{J_{r^{(k)}}(1,1)} \int\limits_{\mathbb{R}^2} J_{r^{(k)}}(\xi_1, \xi_2) (\xi^2)^{[r^{(k)}]} |F[u](\xi)|^2 d\xi_1 d\xi_2 =$$

$$\frac{1}{J_{r^{(k)}}(1,1)} \int\limits_{\mathbb{R}^2} \left(\int\limits_{\mathbb{R}^2} \frac{\left|e^{i\xi_1 z_1} - 1\right|^2 \left|e^{i\xi_2 z_2} - 1\right|^2 (\xi^2)^{[r^{(k)}]} |F[u](\xi)|^2}{|z_1|^{1+2r_1^{(k)}} |z_2|^{1+2r_2^{(k)}}} dz_1 dz_2 \right) d\xi_1 d\xi_2.$$

Учитывая, что в последнем интеграле подынтегральное выражение – неотрицательное, можем изменить порядок интегрирования, согласно теореме Фубинни.

$$\int\limits_{\mathbb{R}^2} (\xi^2)^{r^{(k)}} |F[u](\xi)|^2 d\xi_1 d\xi_2 =$$

$$= \frac{1}{J_{r^{(k)}}(1,1)} \int\limits_{\mathbb{R}^2} \left(\int\limits_{\mathbb{R}^2} \frac{\left|e^{i\xi_1 z_1} - 1\right|^2 \left|e^{i\xi_2 z_2} - 1\right|^2 (\xi^2)^{[r^{(k)}]} |F[u](\xi)|^2}{|z_1|^{1+2r_1^{(k)}} |z_2|^{1+2r_2^{(k)}}} d\xi_1 d\xi_2 \right) dz_1 dz_2 =$$

$$= \frac{1}{J_{r^{(k)}}(1,1)} \times$$

$$\times \int\limits_{\mathbb{R}^2} \left(\int\limits_{\mathbb{R}^2} \frac{\left|\left(e^{i\xi_1 z_1} e^{i\xi_2 z_2} - e^{i\xi_1 z_1} - e^{i\xi_2 z_2} + 1\right)(-i\xi)^{[r^{(k)}]} F[u](\xi)\right|^2}{|z_1|^{1+2r_1^{(k)}} |z_2|^{1+2r_2^{(k)}}} d\xi_1 d\xi_2 \right) dz_1 dz_2$$

Во внутреннем интеграле, используя свойства преобразования Фурье и равенство Парсеваля, получим:

$$\int_{\mathbb{R}^2} (\xi^2)^{r^{(k)}} |F[u](\xi)|^2 d\xi_1 d\xi_2 =$$

$$\frac{1}{J_{r^{(k)}}(1,1)} \int_{\mathbb{R}^2} \left(\int_{\mathbb{R}^2} \frac{\left| \Delta_1(z_1)\Delta_2(z_2) D^{[r^{(k)}]} u(x) \right|^2}{|z_1|^{1+2r_1^{(k)}} |z_2|^{1+2r_2^{(k)}}} dx_1 dx_2 \right) dz_1 dz_2 =$$

$$= \frac{I\left(r^{(k)}, u\right)}{J_{r^{(k)}}(1,1)} < \infty. \tag{2.8}$$

Этим отношение (2.6) доказано.

Обратно покажем, что если функция u принадлежит множеству стоящей в правой части (2.4), то $u \in W_2^{\mathfrak{N}}(\mathbb{R}^2)$. Так как точка $\left[r^{(k)}\right](k = 1, \dots, M)$ принадлежит многограннику $\mathfrak{N}$, то (см., например [6]) с некоторой постоянной $c_0\left(\mathfrak{N}, \left[r^{(k)}\right]\right) > 0$ выполняется неравенство

$$(\xi^2)^{[r^{(k)}]} \leq c_0 \cdot P_{\mathfrak{N}}(\xi), \tag{2.9}$$

Следовательно, из $\sqrt{P_{\mathfrak{N}}(\xi)} F[u](\xi) \in L_2(\mathbb{R}^2)$ получим, что $(-i\xi)^{[r^k]} F[u](\xi) \in L_2(\mathbb{R}^n)$, откуда в силу свойства преобразования Фурье получим, что $D^{[r^k]} u \in L_2(\mathbb{R}^n)(k = 1, \dots, M)$. Ограниченность $I(r^k, u)$ доказывается аналогично, как выше, применяя обратные шаги преобразования.

Покажем эквивалентность норм:

$$\| u \|_{W_2^{\mathfrak{N}}(\mathbb{R}^2)} = \| u \|_{L_2(\mathbb{R}^2)} + \sum_{k=1}^{M} N\left(r^{(k)}, u\right) =$$

$$= \| u \|_{L_2(\mathbb{R}^2)} + \sum_{r^{(k)} = [r^{(k)}]} \| D^{r^{(k)}} u \|_{L_2(\mathbb{R}^2)} + \sum_{[r^k] \neq r^k} I\,(r^k, u)^{\frac{1}{2}}.$$

Используя равенство Парсеваля и равенство (2.8), можем написать

$$\| u \|_{W_2^{\mathfrak{N}}(\mathbb{R}^2)} = \left(\int\limits_{\mathbb{R}^2} |F[u](\xi)|^2 \, d\xi \right)^{\frac{1}{2}}$$

$$+ \sum_{r^{(k)}=[r^{(k)}]} \left(\int\limits_{\mathbb{R}^2} (\xi^2)^{r^{(k)}} \, |F[u](\xi)|^2 d\xi \right)^{\frac{1}{2}} +$$

$$+ \sum_{[r^{(k)}]\neq r^{(k)}} \sqrt{J_{r^{(k)}}(1,1)} \left(\int\limits_{\mathbb{R}^2} (\xi^2)^{r^{(k)}} \, |F[u](\xi)|^2 d\xi \right)^{\frac{1}{2}}.$$

Обозначив через $M_{\mathfrak{N}}$ и $m_{\mathfrak{N}}$, соответственно, наибольшее и наименьшее из чисел $J_{r^{(k)}}(1,1)$, $[r^{(k)}] \neq r^{(k)}$ и 1. Так как очевидно, что для любого набора неотрицательных чисел $a_j, j = 1,2,\cdots,k$ и числа $p \geq 0$ с некоторыми констатами $c_1(p,k) > 0$ и $c_2(p,k) > 0$, не зависящие от самих чисел a_j, выполняется неравенство

$$c_1(p,k)(a_1 + \cdots + a_k)^p \leq a_1^p + \cdots + a_k^p \leq c_2(p,k)(a_1 + \cdots + a_k)^p, \quad (2.10)$$

можем написать

$$\frac{1}{M_{\mathfrak{N}} c_2\left(\frac{1}{2},3\right)} \cdot \| u \|_{W_2^{\mathfrak{N}}(\mathbb{R}^2)} \leq \left(\int\limits_{\mathbb{R}^2} P_{\mathfrak{N}}(\xi) |F[u](\xi)|^2 d\xi \right)^{\frac{1}{2}} \leq$$

$$\leq \frac{1}{m_{\mathfrak{N}} c_1\left(\frac{1}{2},3\right)} \cdot \| u \|_{W_2^{\mathfrak{N}}(\mathbb{R}^2)}.$$

Что и требовалось доказать.

Лемма 2.2. Для любого вполне правильного многогранника $\mathfrak{N}$, вершины которого принадлежат $\mathbb{Q}_+^2$, $\mathrm{W}_2^{\mathfrak{N}}(\mathbb{R}^2)$, является Банаховым пространством.

Доказательство. Пусть u_n – фундаментальная последовательность функций из $W_2^{\mathfrak{N}}(\mathbb{R}^2)$. Согласно лемме 2.1, последовательность функций $\sqrt{P_{\mathfrak{N}}(\xi)}F[u_n] \in L_2(\mathbb{R}^2)$ будет фундаментальной в $L_2(\mathbb{R}^2)$. Из полноты пространства $L_2(\mathbb{R}^2)$ вытекает существование $u_0 \in L_2(\mathbb{R}^2)$, для которого

$$\lim_{n\to\infty}\int\limits_{\mathbb{R}^2} P_{\mathfrak{N}}(\xi)\left|F[u_n](\xi)-\frac{u_0(\xi)}{\sqrt{P_{\mathfrak{N}}(\xi)}}\right|^2 d\xi =$$
$$=\lim_{n\to\infty}\int\limits_{\mathbb{R}^2}\left|\sqrt{P_{\mathfrak{N}}(\xi)}F[u_n](\xi)-u_0(\xi)\right|^2 d\xi = 0$$

Функция $u(\xi) = u_0(\xi)/\sqrt{P_{\mathfrak{N}}(\xi)}$ принадлежит пространству $L_2(\mathbb{R}^2)$. Нетрудно заметить, что $F^{-1}[u] \in W_2^{\mathfrak{N}}(\mathbb{R}^2)$ и является пределом последовательности u_n в пространстве $W_2^{\mathfrak{N}}(\mathbb{R}^2)$, чем и доказывается полнота пространства $W_2^{\mathfrak{N}}(\mathbb{R}^2)$.

Лемма 2.3. Для любого вполне правильного многогранника $\mathfrak{N} \subset \mathbb{R}^2$, вершины которого принадлежат $\mathbb{Q}_2^+$, множество $\mathrm{C}_0^\infty(\mathbb{R}^2)$ плотно в $\mathrm{W}_2^{\mathfrak{N}}(\mathbb{R}^2)$.

Доказательство. Сначала докажем, что $S(\mathbb{R}^2)$ плотно в $W_2^{\mathfrak{N}}(\mathbb{R}^2)$. Пусть m – такое положительное натуральное число, что вершины многогранника $m\mathfrak{N}$ являются мультииндексами. Так как функции $P_{\mathfrak{N}}(\xi)$ и $\sqrt[m]{P_{m\mathfrak{N}}(\xi)}$ эквивалентны, то утверждение леммы 2.1 остается верным, если вместо $P_{\mathfrak{N}}(\xi)$ написать $\sqrt[m]{P_{m\mathfrak{N}}(\xi)}$. Пусть теперь $u \in W_2^{\mathfrak{N}}(\mathbb{R}^2)$ – произвольная функция, из этого следует, что $\sqrt[2m]{P_{m\mathfrak{N}}(\xi)}\hat{u}(\xi) \in L_2(\mathbb{R}^2)$. Из полноты $S(\mathbb{R}^2)$ в $L_2(\mathbb{R}^2)$ следует существование последовательности $u_n \subset S(\mathbb{R}^2)$, для которого

$$\lim_{n\to\infty}\int\limits_{\mathbb{R}^2}\left|u_n(\xi)-\sqrt[2m]{P_{m\mathfrak{N}}(\xi)}\hat{u}(\xi)\right|^2 d\xi = 0,$$

откуда

$$\lim_{n\to\infty}\int\limits_{\mathbb{R}^2}\sqrt[m]{P_{m\mathfrak{N}}(\xi)}\left|\frac{u_n(\xi)}{\sqrt[2m]{P_{m\mathfrak{N}}(\xi)}}-\hat{u}(\xi)\right|^2 d\xi=0.$$

Так как функции $u_n(\xi)/\sqrt[m]{P_{m\mathfrak{N}}(\xi)}\in S(\mathbb{R}^2)$, то по определению нормы в пространстве $W_2^{\mathfrak{N}}(\mathbb{R}^2)$ последовательность функций $\tilde{u}_n(x)\in S(\mathbb{R}^2)$ сходится в функции u по норме в $W_2^{\mathfrak{N}}(\mathbb{R}^2)$, чем и доказывается полнота пространства $S(\mathbb{R}^2)$ в пространстве $W_2^{\mathfrak{N}}(\mathbb{R}^2)$.

Остается доказать, что $C_0^\infty(\mathbb{R}^2)$ плотно в $S(\mathbb{R}^2)$ по норме $W_2^{\mathfrak{N}}(\mathbb{R}^2)$. Возьмем произвольный, вполне правильный многогранник $\mathfrak{N}'$, вершины которого являются мультииндексами – такими, что $\mathfrak{N}\subset\mathfrak{N}'$. Согласно лемме 1.5, $C_0^\infty(\mathbb{R}^2)$ плотно в $W_2^{\mathfrak{N}'}(\mathbb{R}^2)$ по норме $W_2^{\mathfrak{N}'}(\mathbb{R}^2)$. Пусть $u\in S(\mathbb{R}^2)$ – произвольная функция. Рассмотрим функцию u как функцию из $W_2^{\mathfrak{N}'}(\mathbb{R}^2)$. Существует последовательность $u_n\in C_0^\infty$ – такая, что u_n сходится к u по норме $W_2^{\mathfrak{N}'}(\mathbb{R}^2)$. Так как с некоторой постоянной $c>0$ выполняется $\| u \|_{W_2^{\mathfrak{N}}(\mathbb{R})^2}\leq c\,\| u \|_{W_2^{\mathfrak{N}'}(\mathbb{R}^2)}$, то отсюда получаем, что при $n\to\infty$, u_n сходится к u также и по норме $W_2^{\mathfrak{N}}(\mathbb{R}^2)$, чем и завершается доказательство леммы.

3. Следы фукций из мультианизотропных пространств Соболева

В $\mathbb{R}^2$ рассмотрим вполне правильный многогранник $\mathfrak{N}_0$, главные вершины которого являются мультииндексы $\alpha^{(k)}(k=1,2,\cdots,M_0)$. Через $\mathfrak{N}$ обозначим трехмерный, вполне правильный многогранник с вершинами $r^{(k)}=\left(\alpha_1^{(k)},\alpha_2^{(k)},0\right)$ и $r^{(M)}=(0,0,l_3)\in\mathbb{Z}_+^3$, где $M=M_0+1$. $\mathfrak{N}$ из себя представляет пирамиду с основанием $\mathfrak{N}_0$ и с вершиной $r^{(M)}$. Пусть для $x=(x_1,x_2,x_3)$, $x'=(x_1,x_2)$. Согласно обозначениям $P_{\mathfrak{N}_0}(\xi')=1+\sum_{k=1}^{M_0}\left({\xi'}^2\right)^{r^{(k)}}$, $P_{\mathfrak{N}}(\xi)=P_{\mathfrak{N}_0}(\xi')+\xi_3^{2l_3}$.

Теорема 3.1. Пусть $f \in W_2^{\mathfrak{N}}(\mathbb{R}^3)$ – произвольная функция, s – целое число, удовлетворяющее неравенствам $0 \le s < l_3$. Тогда для почти всех $a \in \mathbb{R} D_{x_3}^s f|_{x_3=a} \in W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$, причем с некоторой постоянной $\gamma_1(s, \mathfrak{N}) > 0$, не зависящая от функции f и числа a, выполняется следующая оценка:

$$\| D_{x_3}^s f|_{x_3=a} \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \le \gamma_1 \| f \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)}. \tag{3.1}$$

Доказательство. Сначала докажем оценку (3.1) для любой функции $u \in C_0^\infty(\mathbb{R}^3)$. По определению эквивалентной нормы пространства $W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$, имеем

$$\| D_{x_3}^s u|_{x_3=a} \|^2_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} = \int_{\mathbb{R}^2} P_{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\xi') \left| F_{x'}\left[D_{x_3}^s u\right](\xi', a)\right|^2 d\xi'. \tag{3.2}$$

Из определения преобразования Фурье имеем, что

$$F_{x'}\left[D_{x_3}^s u\right](\xi', x_3) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} (i\xi_3)^s F[u](\xi', \xi_3) e^{ix_3\xi_3} d\xi_3.$$

Откуда

$$F_{x'}\left[D_{x_3}^s u\right](\xi', a) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} (i\xi_3)^s F[u](\xi', \xi_3) e^{ia\xi_3} d\xi_3 =$$

$$= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{+\infty} \frac{(i\xi_3)^s e^{ia\xi_3}}{\sqrt{P_{\mathfrak{N}}(\xi)}} \cdot \sqrt{P_{\mathfrak{N}}(\xi)} \cdot F[u](\xi', \xi_3) d\xi_3.$$

Применяя неравенство Гелдера, получим

$$\left|F_{x'}\left[D_{x_3}^s u\right](\xi', a)\right|^2$$
$$\leq \frac{1}{2\pi} \int_{-\infty}^{+\infty} \frac{\xi_3^{2s}}{P_{\mathfrak{N}_0}(\xi') + \xi_3^{2l_3}} d\xi_3 \cdot \int_{-\infty}^{+\infty} P_{\mathfrak{N}}(\xi) |F[u](\xi', \xi_3)|^2 d\xi_3. \tag{3.3}$$

Рассмотрим первый интеграл. Так как числа l_3 и s целые, то $2l_3 - 2s > 1$, следовательно, последний является сходящимся интегралом. Производя в нем замену переменной $\tau = P_{\mathfrak{N}_0}(\xi')^{-\frac{1}{2l_3}} \xi_3$, получим

$$\int_{-\infty}^{+\infty} \frac{\xi_3^{2s}}{P_{\mathfrak{N}_0}(\xi') + \xi_3^{2l_3}} d\xi_3 = \frac{1}{P_{\mathfrak{N}_0}^{1 - \frac{s}{l_3} - \frac{1}{2l_3}}(\xi')} \cdot \int_{-\infty}^{+\infty} \frac{\tau^{2s}}{1 + \tau^{2l_3}} d\tau. \tag{3.4}$$

Использую неравенство (1.4) и учитывая (3.3) и (3.4), можно выбрать константу

$$\gamma_1(s, l_3, \mathfrak{N}_0) = \frac{c_1\left(1 - \frac{s}{l_3} - \frac{1}{2l_3}, M_0 + 1\right)}{2\pi} \cdot \int_{-\infty}^{\infty} \frac{\tau^{2s}}{1 + \tau^{2l_3}} d\tau,$$

чтобы выполнялось неравенство

$$\left|F_{x'}\left[D_{x_3} u\right](\xi', a)\right|^2 \leq \gamma_1 \cdot \frac{\int_{-\infty}^{+\infty} P_{\mathfrak{N}}(\xi) |F[u](\xi', \xi_3)|^2 d\xi_3}{P_{\left(1 - \frac{s}{l_3} - \frac{1}{2l_3}\right)\mathfrak{N}_0}(\xi')}. \tag{3.5}$$

Подставляя оценку (3.5) в равенство (3.2), получим неравенство (3.1) для функции u.

Пусть теперь $f \in W_2^{\mathfrak{N}}(\mathbb{R}^3)$ – произвольная функция, а $f_n \subset C_0^\infty(\mathbb{R}^3)$ – последовательность функций, сходящей к f по норме $W_2^{\mathfrak{N}}(\mathbb{R}^3)$. Так как последовательность функций f_n сходится к f также по норме в

$L_2(\mathbb{R}^3)$, то можно выделить подпоследовательность f_n^* последовательности f_n, – такую, что для почти всех $a \in \mathbb{R}$ последовательность $f_n^*|_{x_3=a}$ сходится к $f|_{x_3=a}$ по норме $L_2(\mathbb{R}^2)$. Для таких a и для $\forall n, m \in \mathbb{N}$, согласно неравенству (3.1), имеем

$$\| D_{x_3}^s f_n^*|_{x_3=a} - D_{x_3}^s f_m^*|_{x_3=a} \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \le \gamma_1 \| f_n^* - f_m^* \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)}.$$

Из фундаментальности последовательности f_n^* в пространстве $W_2^{\mathfrak{N}}(\mathbb{R}^3)$ следует фундаментальность последовательности $D_{x_3}^s f_n^*|_{x_3=a}$ в пространстве $W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$. Следовательно, в силу полноты последней существует функция $f_a \in W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$, к которой сходится эта последовательность. Переходя к пределу, когда $n \to \infty$ в неравенстве

$$\| D_{x_3}^s f_n^*|_{x_3=a} \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \le \gamma_1 \| f_n^* \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)},$$

получаемое из (3.1), имеем что

$$\| f_a \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \le \gamma_1 \| f \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)}.$$

Так как f_a есть предел последовательности $D_{x_3}^s f_n^*|_{x_3=a}$ также по норме в $L_2(\mathbb{R}^2)$, то по единственности предела в пространстве $L_2(\mathbb{R}^2)$ получим, что f_a совпадает с $D_{x_3}^s f|_{x_3=a}$, чем и завершается доказательство.

Теорема 3.2. Пусть $\mathrm{f} \in \mathrm{W}_2^{\mathfrak{N}}(\mathbb{R}^3)$ и s – целое число, удовлетворяющее неравенствам $0 \le \mathrm{s} < \mathrm{l}_3$. Тогда для каждого $\mathrm{a} \in \mathbb{R}$, если обозначить через $\mathfrak{A}$ множество всех тех ε, для которых $\mathrm{D}_{\mathrm{x}_3}^{\mathrm{s}} \mathrm{f}|_{\mathrm{x}_3=\mathrm{a}+\varepsilon} \in$

$W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$, то существует функция $\varphi_s \in W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$ – такая, что выполняются следующие соотношения:

$$\lim_{\substack{\varepsilon\to 0\\ \varepsilon\in\mathfrak{A}}} \| D_{x_3}^s f|_{x_3=a+\varepsilon} - \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} = 0, \tag{3.6}$$

$$\| \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \le \gamma_2 \| f \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)}, \tag{3.7}$$

где $\gamma_2(\mathrm{s},\mathfrak{N}) > 0$ константа, независящая от функции f и числа a. Причем если $\mathrm{D}_{\mathrm{x}_3}^{\mathrm{s}}\mathrm{f}|_{\mathrm{x}_3=\mathrm{a}} \in W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$, то φ_s и $\mathrm{D}_{\mathrm{x}_3}^{\mathrm{s}}\mathrm{f}|_{\mathrm{x}_3=\mathrm{a}}$ совпадают.

Доказательство. Для произвольных $\varepsilon_1, \varepsilon_2 \in \mathfrak{A}$ рассмотрим функцию $g(x) = f(x', x_3+\varepsilon_1) - f(x', x_3+\varepsilon_2)$. Очевидно, что $g \in W_2^{\mathfrak{N}}(\mathbb{R}^3)$ и относительно функции g применима теорема 3.2, согласно которой выполняется следующее неравенство:

$$\| D_{x_3}^s f(x', a+\varepsilon_2) - D_{x_3}^s f(x', a+\varepsilon_2) \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \le$$
$$\le \gamma_1 \| f(x', x_3+\varepsilon_1) - f(x', x_3+\varepsilon_2) \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)}. \tag{3.8}$$

Пусть ε_1 и ε_2, независимо друг от друга, стремятся к нулю, тогда в силу непрерывности, в целом, функции f – правая часть неравенства (3.8), следовательно, и левая часть будут стремиться к нулю, откуда, ввиду полноты пространства $W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$, вытекает существование функции φ_s, для которого выполняется соотношение (3.6). Далее, согласно теореме 3.1 и в силу (3.6), будем иметь

$$\| \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} = \lim_{\substack{\varepsilon\to 0\\ \varepsilon\in\mathfrak{A}}} \| D_{x_3}^s f|_{x_3=a+\varepsilon} \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \le$$
$$\gamma_1 \| f \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)},$$

чем и доказывается неравенство (3.7) с константой $\gamma_2 = \gamma_1$. Наконец, последнее утверждение теоремы вытекает из единственности предела в пространстве $W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$.

Теорема 3.3. По заданному набору функций $\varphi_s \subset S(\mathbb{R}^2)$, где $s = 0,1,\dots,l_3-1$, и числа $a \in \mathbb{R}$ можно построить функцию $f \in S(\mathbb{R}^3)$, удовлетворяющая следующим свойствам. Для $\forall s = 0,1,\dots,l_3-1$

$$D_{x_3}^s f|_{x_3=a} = \varphi_s, \tag{3.9}$$

$$\lim_{\varepsilon\to 0} \| D_{x_3}^s f|_{x_3=a+\varepsilon} - \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} = 0, \tag{3.10}$$

$$\| f \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)} \le \gamma_3 \sum_{s=0}^{l_3-1} \| \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)}, \tag{3.11}$$

где $\gamma_3(\mathfrak{N}) > 0$ константа, независящая от набора функций φ_s и числа a.

Доказательство. Для доказательства достаточно построить функцию $f_0 \in S(\mathbb{R}^3)$, удовлетворяющую соотношениям (3.9) – (3.11) в случае $a = 0$ с константой $\gamma_3(\mathfrak{N}) > 0$, не зависящая от набора функций φ_s. Действительно, f можно определить, взяв $f(x_1,x_2,x_3) = f_0(x_1,x_2,x_3-a)$, так как будем иметь $D_{x_3}^s f|_{x_3=a} = D_{x_3}^s f_0|_{x_3=0}$, $\| f \|_{W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)} = \| f_0 \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)}$ при любом $a \in \mathbb{R}$.

Для фиксированного значения s рассмотрим функцию $\psi_s \in C_0^\infty(\mathbb{R})$ – такую, что $\psi_s^{(j)}(0) = 0$ для $\forall j < l_3, j \neq s$ и $\psi_s^{(s)}(0) = 1$. Рассмотрим функцию

$$f_s(x_1,x_2,x_3) = F_{\xi'}^{-1}\left[\psi_s\left(x_3 \cdot P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}\right) P_{\mathfrak{N}_0}(\xi')^{-\frac{s}{2l_3}} F[\varphi_s](\xi')\right], \tag{3.12}$$

принадлежащую пространству $S(\mathbb{R}^3)$. Сначала покажем, что построенная таким образом функция $f_s(x)$ удовлетворяет следующим условиям.

$$D_{x_3}^{j} f_s(x)|_{x_3=0} = 0, \text{ для } \forall j < l_3, j \neq s, \tag{3.13}$$

$$D_{x_3}^{s} f_s(x)|_{x_3=0} = \varphi_s(x'), \tag{3.14}$$

$$\| f_s \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)} \leq \gamma_{3,s} \cdot \| \varphi^s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)}, \tag{3.15}$$

где $\gamma_{3,s}(\mathfrak{N}, s) > 0$ – константа, независящая от функции φ_s.

Докажем выполнение соотношений (3.13) и (3.14).

$$D_{x_3}^{j} f_s(x) = F_{\xi'}^{-1}\left[\psi_s^{(j)}\left(x_3 \cdot P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}\right) P_{\mathfrak{N}_0}(\xi')^{\frac{j-s}{2l_3}} F[\varphi_s](\xi')\right],$$

Следовательно, при $\forall j < l_3, j \neq s$

$$D_{x_3}^{j} f_s(x)|_{x_3=0} = \psi_s^{(j)}(0)\varphi_s(x') = 0,$$

Аналогично, при $j = s$ имеем

$$D_{x_3}^{s} f_s|_{x_3=0} = \psi_s^{(s)}(0)\varphi_s(x') = \varphi_s(x').$$

Докажем соотношение (3.15). Для преобразования Фурье-функции $f_s(x_1, x_2, x_3)$ имеем

$$F[f_s](\xi) = \int\limits_{\mathbb{R}^3} e^{-ix\xi} F_{\xi'}^{-1}\left[\psi_s\left(x_3 P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}\right) F[\varphi_s](\xi') P_{\mathfrak{N}_0}(\xi')^{-\frac{s}{2l_3}}\right] dx =$$

$$\int_{\mathbb{R}^1} e^{-ix_3\xi_3}\left(\int_{\mathbb{R}^2} e^{-ix'\xi'}\, F_{\xi'}^{-1}\left[\psi_s\left(x_3 P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}\right) F[\varphi_s](\xi') P_{\mathfrak{N}_0}(\xi')^{-\frac{s}{2l_3}}\right] dx_1 dx_2\right) dx_3$$
$$=$$
$$\int_{\mathbb{R}^1} e^{-ix_3\xi_3}\,\psi_s\left(x_3 P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}\right) P_{\mathfrak{N}_0}(\xi')^{-\frac{s}{l_3}} F[\varphi_s](\xi') dx_3$$
$$F[\varphi_s](\xi') P_{\mathfrak{N}_0}(\xi')^{-\frac{s}{2l_3}} \int_{\mathbb{R}^1} e^{-x_3\xi_3}\,\psi_s\left(x_3 P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}\right) dx_3.$$

В последнем интеграле, производя замену $z = x_3 P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}$, получим

$$F[f_s](\xi') = F[\varphi_s](\xi') P_{\mathfrak{N}_0}(\xi')^{-\frac{s+1}{2l_3}} \int_{\mathbb{R}^1} e^{-\frac{z\xi_3}{P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}}}\,\psi_s(z) dz =$$
$$F[\varphi_s](\xi') P_{\mathfrak{N}_0}(\xi')^{-\frac{s+1}{2l_3}} F[\psi_s]\left(\frac{\xi_3}{P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}}\right).$$

Отсюда

$$\| f_s \|^2_{W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)} = \int_{\mathbb{R}^3} P_{\mathfrak{N}}(\xi) |F[f_s](\xi)| d\xi =$$
$$\int_{\mathbb{R}^3} \left(P_{\mathfrak{N}_0}(\xi') + \xi_3^{2l_3}\right) |F[\varphi_s](\xi')|^2 P_{\mathfrak{N}_0}(\xi')^{-\frac{s+1}{l_3}} \left| F[\psi_s]\left(\frac{\xi_3}{P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}}\right)\right| d\xi =$$
$$\int_{\mathbb{R}^2} |F[\varphi_s](\xi')|^2\, P_{\mathfrak{N}_0}(\xi')^{-\frac{s+1}{l_3}} \left(\int_{\mathbb{R}^1} \left(P_{\mathfrak{N}_0}(\xi')\right.\right.$$
$$\left.\left. + \xi_3^{2l_3}\right) \left| F[\psi_s]\left(\frac{\xi_3}{P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}}\right)\right| d\xi_3\right) d\xi'.$$

Во внутренном интеграле, производя замену $\tau = \xi_3 / P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}}$ и учитывая, что, согласно (2.10), с постоянной $c_1\left(1 - \frac{1}{2l_3} - \frac{s}{l_3}, M_0 +\right.$

$1\Big) > 0$ выполняется неравенство $P_{\mathfrak{N}_0}^{1-\frac{1}{2l_3}-\frac{s}{l_3}}(\xi') \le c_1^{-1} P_{\left(1-\frac{1}{2l_3}-\frac{s}{l_3}\right)\mathfrak{N}_0}(\xi')$ получим:

$$\| f_s \|^2_{W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)} =$$

$$\int_{\mathbb{R}^2} |F[\varphi_s](\xi')|^2\, P_{\mathfrak{N}_0}(\xi')^{-\frac{s+1}{l_3}} \left(\int_{\mathbb{R}^1} \Big(P_{\mathfrak{N}_0}(\xi') \right.$$

$$\left. + \tau^{2l_3} P_{\mathfrak{N}_0}(\xi')\Big)\, |F[\psi_s](\tau)|^2 P_{\mathfrak{N}_0}(\xi')^{\frac{1}{2l_3}} d\tau \right) d\xi' =$$

$$\int_{\mathbb{R}^2} |F[\varphi_s](\xi')|^2\, P_{\mathfrak{N}_0}^{1-\frac{1}{2l_3}-\frac{s}{l_3}}(\xi') \left(\int_{\mathbb{R}^1} (1+\tau^{2l_3})\, |F[\psi_s](\tau)|^2 d\tau \right) d\xi' \le$$

$$\le c_1^{-1} \int_{\mathbb{R}^2} |F[\varphi_s](\xi')|^2\, P_{\left(1-\frac{1}{2l_3}-\frac{s}{l_3}\right)\mathfrak{N}_0}(\xi') \left(\int_{\mathbb{R}^1} (1 \right.$$

$$\left. + \tau^{2l_3})\, |F[\psi_s](\tau)|^2 d\tau \right) d\xi' =$$

$$c_1^{-1} \int_{\mathbb{R}^1} (1+\tau^{2l_3})\, |F[\psi_s](\tau)|^2 d\tau \cdot \| \varphi_s \|^2_{W_2^{\left(1-\frac{1}{2l_3}-\frac{s}{l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)}.$$

Доказательство соотношения (3.15) завершается выбором константы

$$\gamma_{3,s} = \left(c_1^{-1} \int_{\mathbb{R}^1} (1+\tau^{2l_3})\, |F[\psi_s](\tau)|^2 d\tau \right)^{\frac{1}{2}}.$$

Теперь ясно, что функция $f_0(x) := \sum_{s=0}^{l_3-1} f_s(x)$ удовлетворяет соотношениям (3.9) – (3.11) в случае $a = 0$ с константой $\gamma_3 = \max_{0\le s<l_3} \gamma_{3,s}$, причем выполнение соотношения (3.10) легко следует из свойств функции $\psi_s(x)$.

Теорема 3.4. Для любого заданного набора функций $\varphi_s \subset W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$, где $s = 0,1,\dots,l_3 - 1$ и числа $a \in \mathbb{R}$ существует функция $f \in W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)$, удовлетворяющая следующим свойствам:

$$\lim_{\varepsilon\to 0} \| D_{x_3}^s f|_{x_3=a+\varepsilon} - \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} = 0 \text{ для } \forall s = 0,1 \dots, l_3 - 1 \tag{3.16}$$

$$\| f \|_{W_2^{\mathfrak{N}}(\mathbb{R}^3)} \leq \gamma_4 \sum_{s=0}^{l_3-1} \| \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)}, \tag{3.17}$$

где $\gamma_4(\mathfrak{N}) > 0$ – константа, не зависящая от функций φ_s.

Доказательство. Для каждой функции φ_s выберем последовательность $\{\varphi_{s,n}\}_{n=1}^{\infty} \subset S(\mathbb{R}^2)$, сходящуюся к ней по норме в $W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$. Для каждого n обозначим через f_n функцию, полученную, согласно теореме 3.3, относительно набора $\varphi_{s,n}(s = 0,1, \dots, l_3 - 1)$ и числа a. Нетрудно заметить, что относительно набора $\varphi_{s,n} - \varphi_{s,m}(s = 0,1,\dots,l_3 - 1)$ и числа a будет порождаться функция $f_n - f_m$, следовательно, согласно неравенстве (3.11), будем иметь:

$$\| f_n - f_m \|_{W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)} \leq \gamma_3 \sum_{s=0}^{l_3-1} \| \varphi_{n,s} - \varphi_{m,s} \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)}.$$

Из фундаментальности каждой из последовательностей $\{\varphi_{s,n}\}_{n=1}^{\infty}(s = 0,1,\dots,l_3 - 1)$ в пространстве $W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$ следует фундаментальность последовательности f_n в пространстве $W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)$, и по полноте последней существует функция $f \in W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)$, к которой

стремится последовательность f_n в пространстве $W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)$. Докажем, что функция f удовлетворяет соотношениям (3.16) и (3.17). Для каждого s имеем

$$\| D_{x_3}^s f|_{x_3=a+\varepsilon} - \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \leq$$
$$\| D_{x_3}^s f|_{x_3=a+\varepsilon} - D_{x_3}^s f_n|_{x_3=a+\varepsilon} \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} +$$
$$\| D_{x_3}^s f_n|_{x_3=a+\varepsilon} - D_{x_3}^s f_n|_{x_3=a} \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} +$$
$$\| D_{x_3}^s f_n|_{x_3=0} - \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)}.$$

Используя теорему 3.1 и учитывая, что $D_{x_3}^s f_n|_{x_3=a} = \varphi_{s,n}$, можем продолжать следующим образом:

$$\| D_{x_3}^s f|_{x_3=a+\varepsilon} - \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \leq$$
$$\gamma_3 \| f - f_n \|_{W_2^{(l_3,\mathfrak{N}_0}(\mathbb{R}^3))} +$$
$$\| D_{x_3}^s f_n|_{x_3=a+\varepsilon} - D_{x_3}^s f_n|_{x_3=a} \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} +$$
$$\| \varphi_{s,n} - \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)}.$$

Пусть $\sigma > 0$ произвольно. Принимая во внимание сходимость f_n к f по норме в $W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)$ и сходимость $\varphi_{s,n}$ к φ_s по норме в $W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)$, когда $n \to \infty$, можем подобрать значение $n = n_0$ досточно большим, чтобы выполнялось

$$\| D_{x_3}^s f|_{x_3=a+\varepsilon} - \varphi_s \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \leq$$
$$\sigma + \| D_{x_3}^s f_{n_0}|_{x_3=a+\varepsilon} - D_{x_3}^s f_{n_0}|_{x_3=a} \|_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)}.$$

Из свойства (3.10) функции f_{n_0} получим

$$\overline{\lim_{\varepsilon\to 0}} \parallel D^s_{x_3} f|_{x_3=a+\varepsilon} - \varphi_s \parallel_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} \le \sigma.$$

Так как последнее соотношение имеет место для произвольного $\sigma > 0$, наконец будем иметь, что

$$\lim_{\varepsilon\to 0} \parallel D^s_{x_3} f|_{x_3=a+\varepsilon} - \varphi_s \parallel_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} =$$
$$\overline{\lim_{\varepsilon\to 0}} \parallel D^s_{x_3} f|_{x_3=a+\varepsilon} - \varphi_s \parallel_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)} = 0.$$

Выполнение соотношения (3.17) с константой $\gamma_4 = \gamma_3$ сразу следует из неравенства

$$\parallel f_n \parallel_{W_2^{(l_3,\mathfrak{N}_0)}(\mathbb{R}^3)} \le \gamma_3 \sum_{s=0}^{l_3-1} \parallel \varphi_{s,n} \parallel_{W_2^{\left(1-\frac{s}{l_3}-\frac{1}{2l_3}\right)\mathfrak{N}_0}(\mathbb{R}^2)}$$

переходя к пределу при $n \to \infty$.

ЛИТЕРАТУРА

1. *Карапетян Г.А*. Интегральное представление и теоремы вложения для n-мерных мультианизотропных пространств с одной вершиной анизотропности // Сиб. Мат. журнал. 2017. Т. 58, 3. СС. 573–590.
2. *Демиденко Г.В*. Пространства Соболева и обобщенные решения. Учебное пособие. Новосибирск: РИЦ НГУ, 2015.
3. *Бесов О.В., Ильин В.П., Никольский С.М.* Интегральные представления функций и теоремы вложения. М.: Наука, 1975.
4. *Никольский С.М.* Приближение функций многих переменных и теоремы вложения. М.: Наука, 1977.
5. *Слободецкий Л.Н.* Обобщенные пространства С.Л. Соболева и их приложение к краевым задачам для дифференциальных уравнений в частных производных//Ленинградский Государственный педагогический институт имени

А.И. Герцена Ученые записки, физико-математический факультет, Л., 1958. Т. 197.
6. *Волевич Л.Р., Гиндикин С.Г.* Метод многогранника Ньютона в теории дифференциальных уравнений в частных производных. М.: Эдиториал УРСС, 2002.
7. *Karapetyan G.A*. Integral representations of functions and embedding theorems for multianisotropic spaces on the plane with one anisotropy vertex // Journal of Contemporary Mathematical Analysis. 2016. Vol. 51, No 6. PP. 269–281.
8. *Karapetyan G.A., Arakelyan M.K.* Estimation of multianisotropic kernels and their application to the embedding theorems. Transactions of A. Razmadze Mathematical Institute. 2017. Vol. 171, No 1. PP. 48–56.
9. *Karapetyan G.A., Petrosyan H.A.* Embedding theorems for multianisotropic spaces with two vertices of anisotropicity. Proceedings of the YSU, Physical and Mathematical Sciences. 2017. Vol 51, No 1. PP. 29 –37.
10. *Karapetyan G.A.* An Integral Representation and Embedding Theorems in the Plane for Multianisotropic Spaces Journal of Contemporary Mathematical Analysis (Armenian Academy of Sciences). 2017. Vol. 52, No. 6. PP. 267–275.
11. *Karapetyan G.A., Arakelyan M.K.* Embedding Theorems for General Multianisotropic Spaces Matematical Notes. 2018. Vol. 104, No. 3. PP. 422–438.
12. *Ghazaryan H.G.* The Newton polyhedron, spaces of differentiable functions and general theory of differential equations // Armenian Journal of Mathematics. 2017. Vol. 9, No. 2. PP. 102–145.

ON TRACES OF FUNCTIONS FROM MULTIANISOTROPIC SOBOLEV SPACES

M. Khachaturyan, A. Hakobyan

Russian-Armenian University

ABSTRACT

In this paper, direct and inverse embedding theorems for different dimensions (trace theorems) are obtained for functions from the multianisotropic Sobolev space $W_2^{\mathfrak{N}}(\mathbb{R}^3)$in the case of one class of completely regular polyhedrons $\mathfrak{N}$.

Keywords: completley regular polyhedron, multianisotropic Sobolev space, trace of a function

УДК 519.21

Поступила: 26.03.2021г.
Сдана на рецензию: 29.03.2021г.
Подписана к печати: 06.04.2021г.

CONVEXITY OF THE PROBABILITY DENSITY FUNCTION OF THE DISTANCE BETWEEN TWO POINTS IN A COMPACT SUBSET OF R^1

V. Mnatsakanyan

Russian-Armenian University

v.mnatsakanyan95@gmail.com

ABSTRACT

The paper deals with the classical problems of stochastic tomography: obtaining information about a convexity of the probability density function. In this paper, we obtain that the probability density function of the distance of two uniformly independent distributed points in two segments of a line is not convex (note, that for a segment the function is convex). Also,the probability density function is not non-increasingwhen the midpoint is outside of the segments.

Subclass: 53C65, 53C60, 31A10

Keywords: convexfunction, uniformly distributed random points, non-increasing.

1. Introduction

For two segments in R^1 the length of the first segment we denote by A, the length of the second segment by B and the distance between segments by C.

In this article, we study the monotony and convexity of the density distribution function of the distance of the two random independent

uniformly distributed points in a segment and in two separate segments in a line. Wecalculate the cumulative distribution function and the probability density function of the distance for each case.

The main results are:

Theorem 1. The probability density function of the square of distance of the two random independent points in the two segments in a line is not convex.

Theorem 2. The probability density function of the square of distance of the two random independent points in the two segments with length A and B in a line with the distance of C is not non-increasing, if and only if the midpoint of A+C+B is not included in any of the segment.

The theorems can be used in the epidemiology [1] and in the random networks [2].

2. Proofs of Theorem 1 and 2

First we need to prove the following lemma.

Lemma 1. The probability density function of the square of the distance between the two random independent points in the segment is decreasing and convex.

Proof 1

If the length of the segment is A then the cumulative distribution function of the square of the distance will be $0 \leq \sqrt{t} \leq A$

$$F(t) = \frac{2}{A^2} \int_0^{\sqrt{t}} \int_0^{t_2} dt_1 dt_2 + \int_{\sqrt{t}}^{A} \int_{t_2-\sqrt{t}}^{t_2} dt_1 dt_2 = \frac{2A\sqrt{t}-t}{A^2} \qquad (1.1)$$

Hence density function will be

$$f(t) = \frac{A-\sqrt{t}}{A^2\sqrt{t}} \qquad (1.2)$$

The derivative of the density function will be negative

$$f'(t) = -\frac{1}{2At^{1.5}} < 0 \quad (1.3)$$

Hence the probability density function will be non-increasing

The second derivative will be positive

$$f''(t) = \frac{3}{4At^{1.5}} > 0 \quad (1.3)$$

Hence the probability density function is convex.

Proofs of Theorem 1 and 2:

We assume, without loss of generality, that $A \geq B$

We can study all 5 cases

- $A > B > C; A > B + C$
- $A > B > C; A < B + C$
- $A > C > B; A > B + C$
- $A > C > B; A < B + C$
- $C > A > B$

For each case we can calculate the density function and its derivatives

1. A>B>C; A>B+C

In this case the midpoint is in the insight of the A segment.

$$F(t) = \begin{cases} \frac{2\left((A+B)\sqrt{t}-t\right)}{(A+B)^2}; 0 < \sqrt{t} \leq C \\ \frac{2}{(A+B)^2}\left((A+B-C)\sqrt{t}+\frac{C^2-t}{2}\right); C < \sqrt{t} \leq B \\ \frac{2}{(A+B)^2}\left((A-C)\sqrt{t}+\frac{C^2+B^2}{2}\right); B < \sqrt{t} \leq B+C \\ \frac{2}{(A+B)^2}\left((A+B)\sqrt{t}-BC-\frac{t}{2}\right); B+C < \sqrt{t} \leq A \\ \frac{2}{(A+B)^2}\left(B\sqrt{t}-BC+\frac{A^2}{2}\right); A < \sqrt{t} \leq A+C \\ -\frac{(C-\sqrt{t})(2A+2B+C-\sqrt{t})}{(A+B)^2}; A+C < \sqrt{t} \leq A+B+C \end{cases} \quad (2.1)$$

Density function will be

$$f(t) = \begin{cases} \frac{A+B-2\sqrt{t}}{(A+B)^2\sqrt{t}}; 0 < \sqrt{t} \leq C \\ \frac{(A+B-C-\sqrt{t})}{(A+B)^2\sqrt{t}}; C < \sqrt{t} \leq B \\ \frac{A-C}{(A+B)^2\sqrt{t}}; B < \sqrt{t} \leq B + C \\ \frac{A+B-\sqrt{t}}{(A+B)^2\sqrt{t}}; B + C < \sqrt{t} \leq A \\ \frac{B}{(A+B)^2\sqrt{t}}; A < \sqrt{t} \leq A + C \\ \frac{A+B+C-\sqrt{t}}{(A+B)^2\sqrt{t}}; A + C < \sqrt{t} \leq A + B + C \end{cases} \quad (2.2)$$

As we can see the probability density function is continuous.
The first derivative of the probability density function is

$$f'(t) = \begin{cases} -\frac{1}{2(A+B)t^{1.5}}; 0 < \sqrt{t} < C \\ -\frac{(A+B-C)}{2(A+B)^2t^{1.5}}; C < \sqrt{t} < B \\ -\frac{A-C}{2(A+B)^2t^{1.5}}; B < \sqrt{t} < B + C \\ -\frac{1}{2(A+B)t^{1.5}}; B + C < \sqrt{t} < A \\ -\frac{B}{2(A+B)^2t^{1.5}}; A < \sqrt{t} < A + C \\ -\frac{A+B+C}{2(A+B)^2t^{1.5}}; A + C < \sqrt{t} < A + B + C \end{cases} \quad (2.3)$$

The derivative is negative, which means that the function is decreasing, but the derivative is not continuous and in the point A+C is decreasing.

The second derivative will be

$$f''(t) = \begin{cases} \frac{3}{4(A+B)t^{2.5}}; 0 < \sqrt{t} < C \\ \frac{3(A+B-C)}{4(A+B)^2 t^{2.5}}; C < \sqrt{t} < B \\ \frac{3(A-C)}{4(A+B)^2 t^{2.5}}; B < \sqrt{t} < B+C \\ \frac{3}{4(A+B)t^{2.5}}; B+C < \sqrt{t} < A \\ \frac{3B}{4(A+B)^2 t^{2.5}}; A < \sqrt{t} < A+C \\ \frac{3(A+B+C)}{4(A+B)^2 t^{2.5}}; A+C < \sqrt{t} < A+B+C \end{cases} \quad (2.4)$$

The second derivative is always positive but isn't continuous and in the point A+C is not determined, but the first derivative is decreasing which means that the function is not convex.

In the histogram we can see the case when A=4 B=2 C=1 We can se that function lost the convexity in the points $(B+C)^2$ and $(A+C)^2$.



2. A>B>C; A<B+C

In this case, the midpoint is in the outside of the segments.

The cumulative distribution function will be

$$F(t) = \begin{cases} \frac{2\left((A+B)\sqrt{t}-t\right)}{(A+B)^2}; 0 < \sqrt{t} \le C \\ \frac{2}{(A+B)^2}\left((A+B-C)\sqrt{t} + \frac{C^2-t}{2}\right); C < \sqrt{t} \le B \\ \frac{2}{(A+B)^2}\left((A-C)\sqrt{t} + \frac{C^2+B^2}{2}\right); B < \sqrt{t} \le A \\ \frac{2}{(A+B)^2}\left(-C\sqrt{t} + \frac{C^2+B^2+A^2+t}{2}\right); A < \sqrt{t} \le B + C \\ \frac{2}{(A+B)^2}\left(B\sqrt{t} - BC + \frac{A^2}{2}\right); B + C < \sqrt{t} \le A + C \\ -\frac{(C-\sqrt{t})(2A+2B+C-\sqrt{t})}{(A+B)^2}; A + C < \sqrt{t} \le A + B + C \end{cases} \quad (2.5)$$

We can see, that the Cumulative function is continuous
The probability density function will be

$$f(t) = \begin{cases} \frac{A+B-2\sqrt{t}}{(A+B)^2\sqrt{t}}; 0 < \sqrt{t} \le C \\ \frac{\left(A+B-C-\sqrt{t}\right)}{(A+B)^2\sqrt{t}}; C < \sqrt{t} \le B \\ \frac{A-C}{(A+B)^2\sqrt{t}}; B < \sqrt{t} \le A \\ \frac{\sqrt{t}-C}{(A+B)^2\sqrt{t}}; A < \sqrt{t} \le B + C \\ \frac{B}{(A+B)^2\sqrt{t}}; B + C < \sqrt{t} \le A + C \\ \frac{A+B+C-\sqrt{t}}{(A+B)^2\sqrt{t}}; A + C < \sqrt{t} \le A + B + C \end{cases} \quad (2.6)$$

The derivative of the density function will be

$$f'(t) = \begin{cases} -\frac{1}{(A+B)t^{1.5}}; 0 < \sqrt{t} < C \\ -\frac{(A+B-C)}{2(A+B)^2t^{1.5}}; C < \sqrt{t} < B \\ -\frac{A-C}{2(A+B)^2t^{1.5}}; B < \sqrt{t} < A \\ \frac{C}{2(A+B)^2t^{1.5}}; A < \sqrt{t} < B + C \\ -\frac{B}{2(A+B)^2t^{1.5}}; B + C < \sqrt{t} < A + C \\ -\frac{A+B+C}{2(A+B)^2t^{1.5}}; A + C < \sqrt{t} < A + B + C \end{cases} \quad (2.7)$$

As the derivative can be positive, the density function is not non-increasing

The second derivative will be

$$f''(t) = \begin{cases} \frac{3}{2(A+B)t^{2.5}}; 0 < \sqrt{t} < C \\ \frac{3(A+B-C)}{4(A+B)^2t^{2.5}}; C < \sqrt{t} < B \\ \frac{3(A-C)}{4(A+B)^2t^{2.5}}; B < \sqrt{t} < A \\ -\frac{3C}{4(A+B)^2t^{2.5}}; A < \sqrt{t} < B + C \\ \frac{3B}{4(A+B)^2t^{1.5}}; B + C < \sqrt{t} < A + C \\ \frac{(A+B+C)}{4(A+B)^2t^{2.5}}; A + C < \sqrt{t} < A + B + C \end{cases} \quad (2.8)$$

As when$A < \sqrt{t} < B + C$ the second derivative is negative then the probability density function is not convex.

3. A>C>B; A>B+C

In this case, the midpoint is inside of the segments.

The cumulative distribution function will be

$$F(t)=\begin{cases}\frac{2\left((A+B)\sqrt{t}-t\right)}{(A+B)^2};0<\sqrt{t}\le B\\ \frac{2}{(A+B)^2}\left(A\sqrt{t}-\frac{t}{2}+\frac{B^2}{2}\right);B<\sqrt{t}\le C\\ \frac{2}{(A+B)^2}\left((A-C)\sqrt{t}+\frac{B^2+C^2}{2}\right);C<\sqrt{t}\le B+C\\ \frac{2}{(A+B)^2}\left((A+B)\sqrt{t}-BC-\frac{t}{2}\right);B+C<\sqrt{t}\le A\\ \frac{2}{(A+B)^2}\left(B\sqrt{t}-BC+\frac{A^2}{2}\right);A<\sqrt{t}\le A+C\\ -\frac{(C-\sqrt{t})(2A+2B+C-\sqrt{t})}{(A+B)^2};A+C<\sqrt{t}\le A+B+C\end{cases} \tag{2.9}$$

We can see, that the Cumulative function is continuous
The probability density function will be

$$f(t)=\begin{cases}\frac{A+B-2\sqrt{t}}{(A+B)^2\sqrt{t}};0<\sqrt{t}\le B\\ \frac{A-\sqrt{t}}{(A+B)^2\sqrt{t}};B<\sqrt{t}\le C\\ \frac{A-C}{(A+B)^2\sqrt{t}};C<\sqrt{t}\le B+C\\ \frac{A+B-\sqrt{t}}{(A+B)^2\sqrt{t}};B+C<\sqrt{t}\le A\\ \frac{B}{(A+B)^2\sqrt{t}};A<\sqrt{t}\le A+C\\ \frac{A+B+C-\sqrt{t}}{(A+B)^2\sqrt{t}};A+C<\sqrt{t}\le A+B+C\end{cases} \tag{2.10}$$

The density function is also continuous.
The derivative of density function will be

$$f'(t) = \begin{cases} -\frac{1}{2(A+B)t^{1.5}}; 0 < \sqrt{t} < B \\ -\frac{A}{2(A+B)^2t^{1.5}}; B < \sqrt{t} < C \\ -\frac{A-C}{2(A+B)^2t^{1.5}}; C < \sqrt{t} < B+C \\ -\frac{1}{2(A+B)t^{1.5}}; B+C < \sqrt{t} < A \\ -\frac{B}{2(A+B)^2t^{1.5}}; A < \sqrt{t} < A+C \\ -\frac{A+B+C}{2(A+B)^2t^{1.5}}; A+C < \sqrt{t} < A+B+C \end{cases} \quad (2.11)$$

The derivative of the probability density function is negative which means the function is non-increasing, but it is not continuous which means that it is not convex.

The second derivative will be

$$f''(t) = \begin{cases} \frac{3}{4(A+B)t^{2.5}}; 0 < \sqrt{t} \leq B \\ \frac{3A}{4(A+B)^2t^{2.5}}; B < \sqrt{t} < C \\ \frac{3(A-C)}{4(A+B)^2t^{2.5}}; C < \sqrt{t} < B+C \\ \frac{3}{4(A+B)t^{2.5}}; B+C < \sqrt{t} < A \\ \frac{3B}{4(A+B)^2t^{2.5}}; A < \sqrt{t} < A+C \\ \frac{3(A+B+C)}{4(A+B)^2t^{2.5}}; A+C < \sqrt{t} < A+B+C \end{cases} \quad (2.12)$$

It is positive, but as $f'(t)$ is decreasing in point A+C $f'((A+C)^2 -) > f'((A+C)^2 +)$ it means that the probability distribution function is not convex.

4. A>C>B; A<B+C

In this case, the midpoint is outside of the segments.

The cumulative distribution function will be

$$F(t) = \begin{cases} \frac{2\left((A+B)\sqrt{t}-t\right)}{(A+B)^2}; 0 < \sqrt{t} \leq B \\ \frac{2}{(A+B)^2}\left(A\sqrt{t} - \frac{t}{2} + \frac{B^2}{2}\right); B < \sqrt{t} \leq C \\ \frac{2}{(A+B)^2}\left((A-C)\sqrt{t} + \frac{B^2+C^2}{2}\right); C < \sqrt{t} \leq A \\ \frac{2}{(A+B)^2}\left(-C\sqrt{t} + \frac{C^2+B^2+A^2+t}{2}\right); A < \sqrt{t} \leq B+C \\ \frac{2}{(A+B)^2}\left(B\sqrt{t} - BC + \frac{A^2}{2}\right); B+C < \sqrt{t} \leq A+C \\ -\frac{(C-\sqrt{t})(2A+2B+C-\sqrt{t})}{(A+B)^2}; A+C < \sqrt{t} \leq A+B+C \end{cases} \tag{2.13}$$

The density function is

$$f(t) = \begin{cases} \frac{A+B-2\sqrt{t}}{(A+B)^2\sqrt{t}}; 0 < \sqrt{t} \leq B \\ \frac{A-\sqrt{t}}{(A+B)^2\sqrt{t}}; B < \sqrt{t} \leq C \\ \frac{A-C}{(A+B)^2\sqrt{t}}; C < \sqrt{t} \leq A \\ \frac{\sqrt{t}-C}{(A+B)^2\sqrt{t}}; A < \sqrt{t} \leq B+C \\ \frac{B}{(A+B)^2\sqrt{t}}; B+C < \sqrt{t} \leq A+C \\ \frac{A+B+C-\sqrt{t}}{(A+B)^2\sqrt{t}}; A+C < \sqrt{t} \leq A+B+C \end{cases} \tag{2.14}$$

The derivative of the probability density function will be

$$f'(t) = \begin{cases} -\frac{1}{(A+B)t^{1.5}}; 0 < \sqrt{t} < B \\ -\frac{A}{2(A+B)^2t^{1.5}}; B < \sqrt{t} < C \\ -\frac{A-C}{2(A+B)^2t^{1.5}}; C < \sqrt{t} < A \\ \frac{C}{2(A+B)^2t^{1.5}}; A < \sqrt{t} < B+C \\ -\frac{B}{2(A+B)^2t^{1.5}}; B+C < \sqrt{t} < A+C \\ -\frac{A+B+C}{2(A+B)^2t^{1.5}}; A+C < \sqrt{t} < A+B+C \end{cases} \tag{2.15}$$

As the derivative can be positive, the density function is not non-increasing.

The second derivative will be.

$$f''(t) = \begin{cases} \frac{3}{2(A+B)t^{2.5}}; 0 < \sqrt{t} < B \\ \frac{3A}{4(A+B)^2 t^{2.5}}; B < \sqrt{t} < C \\ \frac{3(A-C)}{4(A+B)^2 t^{2.5}}; C < \sqrt{t} < A \\ -\frac{3C}{4(A+B)^2 t^{2.5}}; A < \sqrt{t} < B + C \\ \frac{3B}{4(A+B)^2 t^{1.5}}; B + C < \sqrt{t} < A + C \\ \frac{(A+B+C)}{4(A+B)^2 t^{2.5}}; A + C < \sqrt{t} < A + B + C \end{cases} \quad (2.16)$$

As if $A < \sqrt{t} < B + C$ the second derivative is negative then the probability density function is not convex.

5. C>A>B;

In this case, the midpoint is outside of the segments.

The cumulative distribution function will be

$$F(t) = \begin{cases} \frac{2\left((A+B)\sqrt{t}-t\right)}{(A+B)^2}; 0 < \sqrt{t} \le B \\ \frac{2}{(A+B)^2}\left(A\sqrt{t} - \frac{t}{2} + \frac{B^2}{2}\right); B < \sqrt{t} \le A \\ \frac{B^2+A^2}{(A+B)^2}; A < \sqrt{t} \le C \\ \frac{2}{(A+B)^2}\left(-C\sqrt{t} + \frac{C^2+B^2+A^2+t}{2}\right); C < \sqrt{t} \le B + C \\ \frac{2}{(A+B)^2}\left(B\sqrt{t} - BC + \frac{A^2}{2}\right); B + C < \sqrt{t} \le A + C \\ -\frac{\left(C-\sqrt{t}\right)\left(2A+2B+C-\sqrt{t}\right)}{(A+B)^2}; A + C < \sqrt{t} \le A + B + C \end{cases} \quad (2.17)$$

The density function is

$$f(t)=\begin{cases}\frac{A+B-2\sqrt{t}}{(A+B)^2\sqrt{t}};0<\sqrt{t}\le B\\ \frac{A-\sqrt{t}}{(A+B)^2\sqrt{t}};B<\sqrt{t}\le A\\ 0;A<\sqrt{t}\le C\\ \frac{\sqrt{t}-C}{(A+B)^2\sqrt{t}};C<\sqrt{t}\le B+C\\ \frac{B}{(A+B)^2\sqrt{t}};B+C<\sqrt{t}\le A+C\\ \frac{A+B+C-\sqrt{t}}{(A+B)^2\sqrt{t}};A+C<\sqrt{t}\le A+B+C\end{cases} \quad (2.18)$$

The derivative of the probability density function will be

$$f'(t)=\begin{cases}-\frac{1}{(A+B)t^{1.5}};0<\sqrt{t}<B\\ -\frac{A}{2(A+B)^2t^{1.5}};B<\sqrt{t}<A\\ 0;A<\sqrt{t}<C\\ \frac{C}{2(A+B)^2t^{1.5}};C<\sqrt{t}<B+C\\ -\frac{B}{2(A+B)^2t^{1.5}};B+C<\sqrt{t}<A+C\\ -\frac{A+B+C}{2(A+B)^2t^{1.5}};A+C<\sqrt{t}<A+B+C\end{cases} \quad (2.19)$$

As the derivative can be positive, the density function is not non-increasing.

The second derivative will be.

$$f''(t)=\begin{cases}\frac{3}{2(A+B)t^{2.5}};0<\sqrt{t}<B\\ \frac{3A}{4(A+B)^2t^{2.5}};B<\sqrt{t}<A\\ 0;A<\sqrt{t}<C\\ -\frac{3C}{4(A+B)^2t^{2.5}};C<\sqrt{t}<B+C\\ \frac{3B}{4(A+B)^2t^{1.5}};B+C<\sqrt{t}<A+C\\ \frac{(A+B+C)}{4(A+B)^2t^{2.5}};A+C<\sqrt{t}<A+B+C\end{cases} \quad (2.20)$$

As if $B < \sqrt{t} < B + C$ the second derivative is negative then the probability density function is not convex.

As we study all cases the theorem is proven.

3. Conclusion

Our theorems show when the probability density function isn't convex for the square of the distance between the two random independent points in the two segments in the line and when it is not non-increasing. For further research, westudy this for convex and non-convex domains in R^2[3]. We can use this result to check if there is eliminated from the segment any subsegment when we are getting the distance of the two signals in the line.

REFERENCES

1. *Souris, M.* Epidemiology and Geography. Principles, Methods and Tools of Spatial Analysis; Wiley-ISTE: Fort Wayne, IN, USA, 2019.
2. *Moltchanov D.* Distance distribution in random networks. Ad Hoc Netw. 2012, 10. 1146–1166.
3. *Aramyan R.H., Mnatsakanyan V.A.* Conditional moments of the distance distribution two random points in a convex domain in R^2, Proceedings of YSU, Physical and Mathematical Sciences, 2020, v. 54, #1. PP. 3–8.

ФУНКЦИИ ПЛОТНОСТИ ВЕРОЯТНОСТИ РАССТОЯНИЯ МЕЖДУ ДВУМЯ ТОЧКАМИ В ПОДМНОЖЕСТВЕ R^1

В. Мнацаканян

Российско-Армянский Университет

АННОТАЦИЯ

В данной научной статье рассматриваются классические задачи стохастической томографии: получение информации о выпуклости функции плотности

вероятности. В этой статье мы показываем, что функция плотности вероятности расстояния между двумя равномерно независимыми распределенными точками в двух сегментах прямой не является выпуклой и, когда средняя точка находится за пределами сегментов, не является невозрастающей.

Подкласс: 53C65, 53C60, 31A10

Ключевые слова: выпуклая функция, равномерно распределенные случайные точки, невозрастание.

УДК 004.021
Поступила: 03.03.2021 г.
Сдана на рецензию: 05.03.2021г.
Подписана к печати: 11.04.2021г.

RESEARCH AND DEVELOPMENT OF A DEEP LEARNING-BASED LEMMATIZER FOR THE ARMENIAN LANGUAGE

T. Ghukasyan, K. Avetisyan

Russian-Armenian University

tsolak.ghukasyan@rau.am

ABSTRACT

In this work, a lightweight and accurate lemmatizer for the Armenian language is developed, using deep learning. Using COMBO neural network as a basis, several alternatives are proposed to replace memory- and parameter-heavy word embeddings with lightweight subword features. Two modifications to fastText algorithm are presented that are based on subword features only. Another alternative, the average of byte-pair segments' embeddings as word's vector reduces the model's size up to 10 times and the size of embeddings around 100 times, while maintaining the same level of accuracy. The developed lemmatizer's accuracy is around the level of the state-of-the-art for small treebank languages.

Keywords: lemmatization, deep learning, word embeddings.

1. Introduction

Lemmatization is the task of finding the dictionary form of words from a given text, and is a common and important component of many text processing systems. It has a wide range of applications that includes text classification, clustering, sequence labeling, information search, word representation learning etc.

The main goal of lemmatization is to amend the sparsity of lexical data by grouping different surface forms of the same word into one. In that sense it is similar to stemming, however the latter's use is established to be less effective and yields relatively worse results in downstream tasks. Since Armenian is a morphologically rich language and does not have large corpora, the availability of text normalization tools is especially critical for developing accurate text processing systems. For the Armenian language, the research in this field has been very limited. Attempts to solve the task during CoNLL Shared Task 2018 [1] did not produce reasonably accurate results, because of a small training dataset. Since then, the second version of ArmTDP dataset [2] has been released, making possible the development of more accurate lemmatizers. In 2018, an end-to-end parser of Eastern Armenian was released by YerevaNN, demonstrating 88.05% lemmatization accuracy on ArmTDP v2.3 test set.

The approaches to lemmatization can be grouped into 3 categories: dictionary lookup, rule-based, machine learning-based. Nowadays, learning-based methods are the prevailing approach to lemmatization, as evidenced by CoNLL Shared Task 2018, where all top-performing systems use machine learning. One of the problems with the latter systems is that they often use large sets of parameters and can be heavy memory-wise. As a result, these systems cannot be run in environments with limited resources.

This work focuses on developing a lightweight and accurate lemmatizer for the Armenian language, using deep learning. This research analyzes the state-of-the-art for low-resource, high-morphology languages; explores the joint training of lemmatization with other tasks; explores avenues of learning lightweight features instead of parameter – and memory-heavy word embeddings.

2. Models

Efforts on developing lemmatizers can be divided into three categories: dictionary lookup, rule based approaches and machine learning based methods. Machine learning based lemmatizers have several

advantages over dictionary lookup and rules: they have the capability to process out-of-vocabulary forms and resolve ambiguous cases based on context, do not require linguistic expertise, and the developed system can be retrained and ported to another language. With deep learning, it is also possible to leverage unlabeled text corpora to learn more effective features.

Generally, machine learning is used to predict the transformation of input word to obtain its lemma. In existing approaches, that transformation is represented in various forms, which can be categorized into edit scripts and character sequence transduction. [3–5] treated lemmatization as a task of character sequence transduction. Overall, using CoNLL Shared Task 2018 as reference, the top performing systems for small treebanks relied on sequence transduction for lemmatization. With the exception of COMBO [6], these systems were based on an encoder-decoder network. COMBO is a neural network that uses convolutional and recurrent layers to parse sentences, predicting lemma and morpho-syntactic labels. The input sentence is passed to the network as a sequence of word-level tokens. First, feature extraction is performed on each token separately then extracted vector is passed through sentence-level recurrent layers to obtain a contextual representation. The contextual representation is then used as an input for task-specific layers.

COMBO's lemmatizer is a transducer that converts the sequence of input word's characters into the sequence of lemma's characters. The input characters are encoded using trainable continuous embeddings. The contextual representation of the input word is reduced through a dense layer and concatenated to the embedding of each input characters, forming their final vector. Then, 3 layers of dilated convolutions are applied to the sequence of character vectors. The output of the final convolutional layer is a sequence of vectors, which are additionally resized to the size of character vocabulary, using a convolutional layer with 1x1 filters. Finally, softmax is applied to each vector in the sequence to determine the most probable character for it. The architecture of the lemmatizer is illustrated in Figure 1.

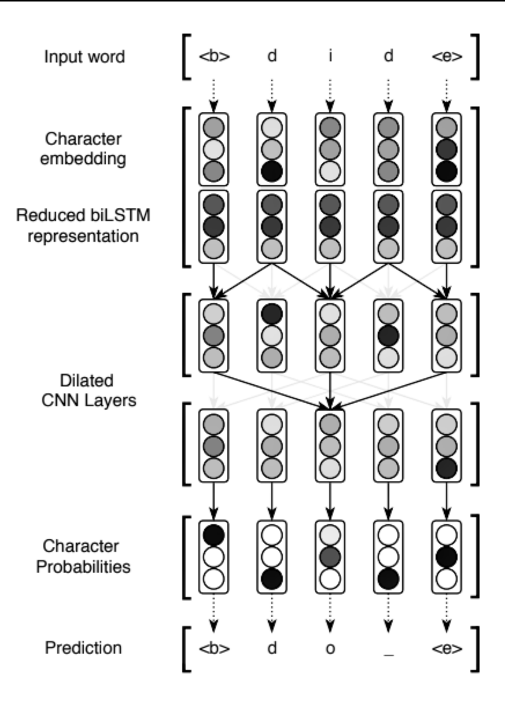


Figure 1. Lemmatizer architecture [6].

For small treebank languages, COMBO's lemmatizer ranked second, only behind the Turku NLP lemmatizer. Considering the large number of parameters in encoder-decoder architectures, extensive training time and the difficulty of hyperparameter tuning, in this work the preference was given to COMBO's neural network for lemmatization.

3. Results and Discussion

In all of experiments the ArmTDP v2.3 test set is used for testing model performance. For supervised learning, the randomly selected 90% of the same dataset's train section is used, while the other 10% is used for validation. For unsupervised learning, a collection of text consisting of 91 million tokens is used. The collection includes Armenian Wikipedia, other

encyclopedia, fiction, as well as news and blog articles from over a dozen different sources.

Models are compared according to total test accuracy, accuracy on out-of-vocabulary words, parameter count. The first baseline is the identity function. As a second baseline, dictionary lookup was chosen, using the train set for constructing the form-lemma dictionary and identity backoff for OOV cases. For forms with multiple possible lemmas, the most common lemma is selected. The third baseline and the starting point for the studied models is the original COMBO lemmatizer, trained separately from other tasks and without self-training. For this baseline, fastText embeddings with vector size 200 are used, trained on a custom collection of texts using SkipGram architecture and character n-grams of size 1 to 3. The word representations used in subsequent experiments were learned using the same set of hyperparameters. The baseline results are reported in Table 1. The overall target accuracy was set to 90%, around the same level as the best models from CoNLL Shared Task 2018 on small treebanks.

Table 1.

Baseline scores

Baseline	Test accuracy	OOV accuracy	Parameter count
identity	52.60	27.34	–
dictionary lookup	74.89	27.34	–
COMBO	87.72	66.42	1.3 x 10e8

The performance of the model in a task can often be improved when it is trained jointly for another related task. In the context of this work, two configurations of joint training are explored: lemmatization and morphological analysis; lemmatization, morphological analysis, and dependency parsing. The experiments demonstrate that the combination of lemmatization and morphological analysis yielded the best results (Table 2).

Table 2.

Joint learning results

Learning configuration	Test Accuracy	OOV Accuracy	Parameter count
lemma	87.72	66.42	1.3 x 10e8
+morph.analysis	**90.77**	**75.25**	**1.3 x 10e8**
+morph.analysis+dep.parsing	90.33	74.75	1.3 x 10e8

In COMBO, fastText embeddings alone constitute as high as 90% of the whole model's parameters. Therefore, when aiming to develop a lightweight lemmatizer, research of smaller word embeddings was selected as the main target. In recent years, several powerful alternatives to word-level embeddings have emerged such as ELMo, Flair and lately BERT. However, they require large training corpora, great computational power, and extensive time to train and tune. BERT particularly is also heavy in parameter count (the publicly available multilingual BERT models have 110 million parameters, around as many as fastText). In this work, two modifications of original fastText [7] model are employed: no-fastText and so-fastText [8]. The first one, considers only word's character n-grams during training and word vector generation, the second one is analogous to the first one, but employs only the n-grams at the end of the word (Figure 2). In two other experiments, word embeddings are replaced by the averaged of their subwords' pre-trained embeddings (Figure 3). Byte-pair encoding [9] is used for obtaining the subwords, and two embedding models: pre-trained vectors from BPEmb project [10] and new vectors trained on our data.

Table 3.

Scores for word embedding alternatives

Lemmatizer	Test Accuracy	OOV Accuracy	Word embedding parameter count	Total parameter count
COMBO+fastText	90.77%	75.25%	1.2 x 10e8	1.3 x 10e8
COMBO+no-fastText	90.36%	73.96%	4.0 x 10e7	5.0 x 10e7

COMBO+so-fastText	89.55%	72.31%	4.0 x 10e7	5.0 x 10e7
COMBO+BPEmb	90.61%	74.92%	2.5 x 10e6	1.2 x 10e7
COMBO+BPE-custom	**90.86%**	**75.62%**	**1.2 x 10e6**	**1.1 x 10e7**

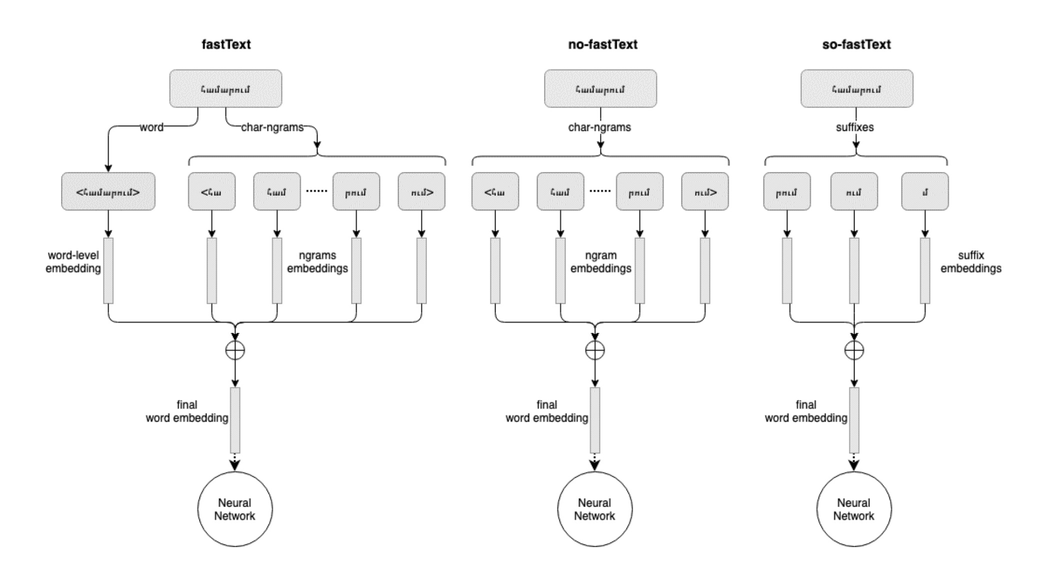


Figure 2. Architecture of fastText, no-fastText, and so-fastText models.

BPEmb is a collection of pre-trained embeddings for byte-pair encoding subwords. They used Wikipedia to learn byte-pair encoding for 275 languages, including Armenian. For representation learning, they used GloVe, which uses word co-occurrence matrix to learn embeddings. Using the unsupervised data, this work develops similar embeddings. Thanks to a small vocabulary size, these subword models are dramatically lighter than conventional word-level embeddings in terms of parameter count and memory. Table 3 provides the results of jointly trained COMBO lemmatizer using fastText and compares it with the results, where fastText is replaced by proposed alternatives.

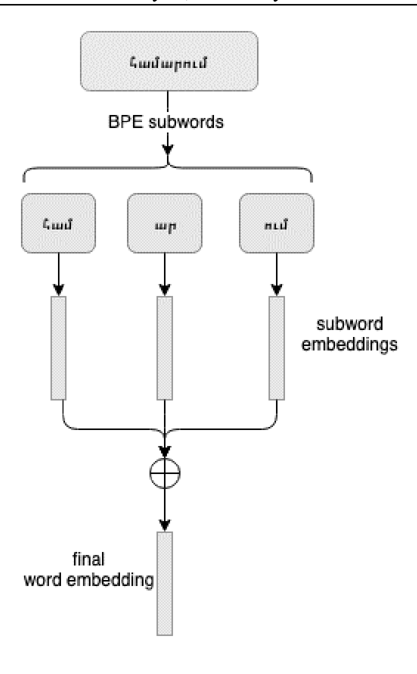


Figure 3. Extraction of BPE-based word embedding.

4. Conclusion

In conclusion, this work explored studied methods of increasing lemmatization accuracy via joint learning and subword-based alternatives to established word embeddings. With no-fastText and so-fastText, a two-fold decrease in the model size is achieved, however the performance is also negatively affected (especially with so-fastText). BPE-based approaches achieved a decrease in embedding parameter count by a factor of almost 100 and a decrease in total parameter count by a factor of around 10. Remarkably, custom BPE embeddings also maintain the same level of accuracy, while requiring a dramatically smaller model. The developed model demonstrates 90.86% accuracy on ArmTDP v2.3 test set, which beats the baseline methods and outperforms the 88.05% of Eastern Armenian End-to-End parser, which was trained and evaluated on the same train and test sets.

REFERENCES

1. *Zeman D. et al.* CoNLL 2018 shared task: Multilingual parsing from raw text to universal dependencies //Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. 2018. PP. 1– 21.
2. *Yavrumyan M.M., Khachatrian H.H., Danielyan A.S., Arakelyan G.D.* Arm TDP: Eastern Armenian Treebank and Dependency Parser. XI International Conference on Armenian Linguistics, Abstracts. Yerevan. 2017.
3. *Dreyer M., Smith J.R., Eisner J.* Latent-variable modeling of string transductions with finite-state methods //Proceedings of the conference on empirical methods in natural language processing. Association for Computational Linguistics, 2008. CC. 1080–1089.
4. *Nicolai G., Kondrak G.* Leveraging Inflection Tables for Stemming and Lemmatization //Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2016. T. 1. CC. 1138–1147.
5. *Eger S., vor der Brück T., Mehler A.* Lexicon-assisted tagging and lemmatization in Latin: A comparison of six taggers and two lemmatization methods // Proceedings of the 9th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH). 2015. CC. 105–113.
6. *Rybak P., Wróblewska A.* Semi-Supervised Neural System for Tagging, Parsing and Lematization //Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies. 2018. CC. 45–54.
7. *Bojanowski P. et al.* Enriching word vectors with subword information //Transactions of the Association for Computational Linguistics. 2017. T. 5. CC. 135–146.
8. Ghukasyan, T., Yeshilbashyan, Y. & Avetisyan, K. Subwords-Only Alternatives to fastText for Morphologically Rich Languages. Program Comput Soft 47, 56–66 (2021). https://doi.org/10.1134/S0361768821010059
9. *Shibata Y. et al.* Byte Pair encoding: A text compression scheme that accelerates pattern matching. – Technical Report DOI-TR-161, Department of Informatics, Kyushu University, 1999.
10. *Heinzerling B. & Strube M.* (2018). BPEmb: Tokenization-free Pre-trained Subword Embeddings in 275 Languages. In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). Miyazaki, Japan. European Language Resources Association (ELRA).

ИССЛЕДОВАНИЕ И РАЗРАБОТКА ЛЕММАТИЗАТОРА ДЛЯ АРМЯНСКОГО ЯЗЫКА НА ОСНОВЕ ГЛУБОКОГО ОБУЧЕНИЯ

Ц.Г. Гукасян, К. Аветисян

Российско-Армянский Университет

АННОТАЦИЯ

В данной работе с использованием глубокого обучения разрабатывается легкий и точный лемматизатор для армянского языка. Используя нейронную сеть «COMBO» в качестве базы для лемматизатора, для замены тяжелых по памяти и количеству параметров предобученных моделей векторов слов предлагается несколько альтернатив на основе подслов. Разрабатываются и тестируются две модификации алгоритма fast Text, использующие только внутренние символьные граммы и суффиксы слова. Другая альтернатива – использование в качестве вектора слова среднее векторов его подслова, полученных в результате byte-pair-кодирования. Последнее позволяет уменьшить размер модели от 1 Гб до несколько Мб, а размер предобученных моделей векторов уменьшается примерно в 100 раз, при этом без потери точности. Точность разработанного лемматизатора находится на уровне лучших решений для языков с небольшим банком деревьев.

Ключевые слова: лемматизация, глубокое обучение, вектора слов.

УДК 004.8

Поступила: 05.04.20212020г.
Сдана на рецензию: 07.04.2021г.
Подписана к печати: 11.04.2021г.

STATIONARITY EFFECT ON TIME SERIES CLUSTERING

K. Gishyan[1], Kh. Smbatyan[2]

[1]University of Bath, UK, [2]Yerevan State University, Armenia

karen.gishyan@bath.edu, smbatyan1729@gmail.com

ABSTRACT

Most of the existing time series clustering applications rely on performing clustering on the original or raw time series data. In this context, we identify a big gap in the literature, as such an approach ignores stationarity, which assumes the mean, variance, and autocorrelation of the series stay constant over time. For two datasets with time series observations which are highly non-stationary and are later transformed to become partially stationary, and stationary, we investigate the performance of the clustering algorithm with three distance metrics and later perform dimensionality reduction with PCA to further evaluate the results. We conclude that although non-stationarity may not worsen clustering performance, making the dataset stationary for highly correlated non-stationary datasets can significantly improve the clustering performance and should be part of the standard time series clustering procedure. Future work may include exploring the performance on multiple datasets with different degrees of multi-collinearity and stationarity for a more comprehensive understanding of the effect.

Keywords: Time Series, Clustering, Stationarity.

1. Introduction

Clustering analysis comprises a set of unsupervised learning methods aiming to find groups of clusters of observations within a data set. The clustering procedure requires determining the number of groups and is used in many fields, such as statistics, biology, and pattern recognition. The greater the homogeneity within the groups and heterogeneity between the groups, the higher the quality of the cluster analysis (Makhlouk, 2018). When performing clustering, two important design criteria are the clustering algorithm and the distance measure. There are three main categories of algorithms: *partitional, density-based*, and *hierarchical*. In this work, we use a popular *K-means* algorithm, and for the *K-means*, the centroid may not be an actual data center (Javed, Lee and Rizzo, 2020). This partition-based algorithm requires that the number of cluster *k* be specified, and given *k*, the algorithm iterates with two stages: (1) calculates centroids, and (2) assigns data points to the closest centroid, until a termination condition (e.g., number of iterations) is met. For improving clustering accuracy, one can perform z-score normalization (Mohamad and Usman, 2013) or experiment with the random seed parameter for selecting clusters with the highest intra-cluster similarity and lowest inter-cluster similarity. Clustering analysis has important applications for time series data, which is a sequence of variable values ordered by time. The applications of time series clustering include financial portfolio building, personalized drug design, and cancer sub-type identification (Javed, Lee and Rizzo, 2020).

This work is part of *Whole time series clustering*, where we apply conventional clustering on discrete objects defined as time series. We work with two datasets that contain an *n*-set of time series variables regarding Masters and PhD education in Armenia and conduct relevant experiments, which we name, *Masters Experiment and PhD Experiment*. Although unsupervised learning techniques have been widely explored, their applications in the time series analysis content have gained particular attention for the recent few years.

2. Literature

Clustering time series data has been used in diverse scientific areas to discover insights from complex datasets, from gene expression data in biology to stock market analysis (Aghabozorgi, Seyed Shirkhorshidi and Ying Wah, 2015). A time series is classified as dynamic data because its values change as a function of time and is also a type of temporal data. Usually clustering such objects leads to a discovery of valuable patterns in time series datasets, and time series clustering is the most used approach as an exploratory and subroutine technique in more involved data mining algorithms such as rule discovery or anomaly detection (Chis, Banerjee and Hassanien, 2009; Aghabozorgi, Seyed Shirkhorshidi and Ying Wah, 2015). Time series clustering is formally defined as follows.

Definition 1 *Time Series clustering. Given a dataset of n time series data* $D=\{F_1,F_2,..,F_n\}$*, the process of unsupervised partitioning of D into* $C=\{C_1,C_2,..,C_k\}$*, in a way that homogeneous time series are grouped based on a similarity measure, is called Time Series clustering.* C_i *is called a cluster, where* $D=\cup_{i=1}^{k} C_i$ *and* $C_i \cap C_j = \emptyset$ *for i 6= j.*

Most of the work in time series clustering is classified into three categories: *Whole time series clustering, Subsequent clustering, and Time point clustering* (Aghabozorgi, Seyed Shirkhorshidi and Ying Wah, 2015).

Aghabozorgi, Seyed Shirkhorshidi and Ying Wah (2015) state that from the existing literature time series clustering has four main components: dimensionality reduction or representation method, distance measurement, clustering algorithm, prototype definition, and evaluation. The dimensionality reduction is especially important when there are memory requirements, and distance calculation among raw data is computationally expensive, and reduction speeds up clustering (Lin et al., 2003; Keogh and Pazzani, 2000). Examples of dimensionality reduction approaches are *Discrete Fourier Transform, Discrete Wavelet Transform, Singular Value decomposition, Discrete Cosine Transform, Singular Value Decomposition, Chebyshev Polynomials, Clipped Data, etc.*

Formally, a representation method is defined.

Definition 2 *Time Series representation. Given a time series data* $F_i=\{f_1,...,F_t,..,F_T\}$, *representation is transforming the time series to another dimensionality reduced* $F_i=\{f_1,...,F_x\}$ *where* $x < T$ *and if two series are similar in the original space, then their representations should be similar in the transformation space too.*

Essentially there are four main types of distance measure in the literature, which are *Shape-based, Comparison-based, Feature-based and model-based*. The *Shape-based* similarity measure is to find the similar time series in time and shape, such as *Euclidean, DTW, LCSS, MVM* measures and is used when working with short time series. *Comparison-based* method is suitable for both short and long time series data and includes measures such as *CDM, AutoCorrelation, Pearson Correlation Coefficient*. *Feature-based, Model-based* measures are suitable for long time series data. The observations of similarity/dissimilarity measures show that most effective distance measures are based on dynamic programming, which, however, are very computationally expensive. For defining cluster centers or prototypes, there are usually three approaches, which imply choosing the medoid sequence of the set, the average sequence of the set, or using local search. In terms of time series clustering algorithms, the most common approaches are *Partitioning, Hierarchical, Grid-based, Model-based, Density-based, and Multi-step Clustering*. One of the most used algorithms for *Partitioning Clustering* are *k-Means, k-Medoids, and Fuzzy c-Means*. *Hierarchical Clustering* uses agglomerative or divisive algorithms for clustering, considers each item as a cluster then gradually merges them bottom-up, unlike divisive, which starts as a single cluster and then splits the cluster to reach clusters with one object. These algorithms cannot adjust the clusters after splitting, so generally, they are combined with other algorithms (Aghabozorgi, Seyed Shirkhorshidi and Ying Wah, 2015). For validating clusters, there are metrics such as *Cluster Similarity Measure, Folkes and Mallow index, Jaccard Score, F-measure, Entropy, and Silhouette Score*.

Limited literature regarding time series clustering addressed non-stationary in their work. Examples of such works are the one of (Xiong and

Yeung, 2002), who apply differencing operations to remove non-stationarity from the time series before proposing a mixture of ARMA models for clustering, and the work of (Wang et al., 2004), who test their proposed clustering approach on both raw time series data as well as the remaining time series after detrending and deseasonalizing.In this section, we also present the methodologies used in this work[1].

2.1. Distance Metrics

For calculating the distance between the clustersin the K-means clustering algorithm, we use *euclidean, Dynamic Time Warping (DTW), and Soft-(DTW)* as our main metrics. *DTW*measures similarity between two temporal sequences, and one of the main advantages is that it can compute distances of time series with different lengths. *Soft-DTW* builds upon DTW and is a differentiable loss function proposed by (Cuturi and Blondel, 2017).

2.2. Silhouette Index

When the number of clusters is not known in advance, the Silhouette Index can be used to obtain the optimal number of clusters. For a given time series x in our dataset in the clusterC_i, we calculate:

- its average dissimilarity with respect to all other time series in the same cluster

$$a(x) = \frac{1}{|C_i - 1|} \times \sum_{y \in C_i, y \neq x} d(x, y)$$

- its average dissimilarity with respect to all other time series in the nearest cluster

$$b(x) = \min_{j, j \neq i} \frac{1}{|C_j|} \sum_{y \in C_j} d(x, y)$$

- the Silhouette value

[1]The code can be found in karen-gishyan/time-series-clustering (github.com).

$$s(x) = \begin{cases} \frac{b(x)-a(x)}{max\{b(x),a(x)\}}, if|C_i| > 1 \\ 0, if|C_i| = 1 \end{cases}$$

- and Silhouette index by

$$Sil(C) = \frac{1}{|C|}\sum_{i=1}^{|C|}\left(\frac{1}{|C_i|}\sum_{x\in C_i} s(x)\right.$$

Silhouette index produces outputs from the range of [-1,1] (Potikyan, 2020), and higher values indicate well-separated clusters.

2.3. Normalization

We use a min-max normalization, which has the following form.

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

where *X* is a series in the dataset.

2.4. Differencing

For making our datasets stationary[2], we use first-order differencing, which is defined as:

$$\Delta Xt = Xt - Xt{-}1$$

2.5. PCA

For the Principal Component Analysis (PCA), *p*-dimensional vectors are projected into a lower *q*-dimensional subspace without losing too much information. The outcome is principal components, which span the subspace in *q* directions (Shalizi, 2012).

[2]The confidence interval for stationarity is taken as 90%.

3. Dataset

The datasets are taken from the Statistical Committee of RA and regard Masters and PhD studies in Armenia[3]. The original Masters dataset and PhD dataset include 60 and 103 columns (time series), respectively. We reduce the number of series to 18, which are the same series for both datasets and can be seen in Table 1.

4. Methodology

We conduct first-order differencing for both Masters and PhD datasets and test stationarity for each time series. To obtain our final dataset, we later identify all non-stationary columns and remove them from both datasets. This leaves us with 9 final time series which are again the same for both datasets. Experimenting with the same set of time series across different stages allows us to have a better understanding of the effect of the conducted technique. We calculate the silhouette index for the original, first-order differenced, and final datasets using *Euclidean, DTW, Soft DTW* distance metrics. We name the relative experiments as*Masters* experiment and *PhD* experiment. We later conduct PCA using *Soft DTW* to observe how clustering performance changes with a reduced set of features to obtain our conclusion.

Stationarity Check

One can remove non-stationarity from the dataset by using methods such as Seasonal Additive/Multiplicative decomposition, STL decomposition, Differencing, etc. We experiment with all three approaches, but as we have yearly data, decomposition methods become less suitable for this task and do not fully remove stationarity, so only the results with first-order

---

[3]The data setscan be obtained from PX-Web - Select table (armstatbank.am) under'Master Programe education institutions by indicators and years'and'Post-Graduation by indicators and years'.

differencing are presented. We conduct an Augmented Dickey Fuller test for the time series in the *Masters* and *PhD* datasets, which tests the null-hypothesis that the series is non-stationary. To make non-stationary series stationary, we do first-order differencing as described in Section 2.5, and present the results in Table 1.

Clustering and Dimensionality Reduction

We use a min-max normalization as described in Section 2.4 for all the time series variables in *Masters* and *PhD* datasets, which scales them from 0 to 1. This method decreases the outlier effect in distance calculation and addresses the negative observations resulted from first-order differencing. To select the best *k* for clustering, we iterate over the *k* cluster parameter ranging 1 to 10, perform a *k-means* clustering, and calculate the *silhouette score* for each *k*. This procedure is repeated for both *Masters* and *PhD* experiments. Later we conduct PCA on the original, first-order differenced, and final datasets across both experiments. As a distance metric, *Soft DTW* is used.

5. Results

Our results are summarized in five tables. We reduce the datasets to include the same set of time series features for both Masters and PhD datasets for comparability and better evaluation. The results from Table 1 show that even though the original PhD dataset contains four stationary variables compared to Masters one, on average the PhD dataset is less stationary than the Masters dataset (closer the P-value is to 1, the less stationary the series is), which can be seen by looking at the *P*-value results of the *Original* column for both datasets. Table 5 shows how correlated on average the datasets are. We see that the Masters dataset is extremely correlated on average, which is another problem to clustering as shown by (Sambandam, 2003) and should be handled. We notice that differencing

helps to decrease average correlation for a highly correlated non-stationary dataset, while has little impact on a low correlated non-stationary dataset. We also observe that stationarity/non-stationarity does not yet imply high/low correlation. We conduct differencing, which makes the majority of the variables stationary under 90% confidence interval, but the impact in terms of stationarizing the series is more for the Masters dataset. To obtain our final dataset, we take the union of non-stationary variables from both datasets, and drop from each, making sure the final Masters and PhD datasets again have the same time series set, and the final set for each dataset contains 9 time series. We consider differencing for each distance metric and each dataset.If we compare the performance of the original series across Table 2 and Table 3, we see that the clustering results are higher for all distance metrics for the PhD experiment, thus we cannot claim that *K-Means* for non-stationary dataset performs worse than on a stationary dataset.However, we are less interested in the performance of the original datasets and more interested in seeing whether making them stationary improves performance. By looking at Table 2, we see that except for *Soft-DTW* for $k = 7$, first-order differencing either keeps the same or improves clustering performance with *Euclidean, DTW*, and *Soft-DTW* distance metrics. From the results of Table 3 for the PhD experiment, although the mixed results, first-order differencing is not very effective. With the *Euclidean* metric, differencing mostly impairs results, for *DTW*, differencing improves performance for $k = 2$ and $k = 3$, but there is a reduced performance for the remaining clusters. For *Soft DTW*, except for clusters $k = 3$ and $k = 6$, *Soft DTW* improves performance. For the final datasets for both experiments, we mostly observe a decrease in performance compared to the first-order differenced datasets.

Table 1: Stationarity for two datasets.

Time Series	Stationarity (P-value)			
	Masters Dataset		PhD Dataset	
	Original	First-Order Differenced	Original	First-Order Differenced
Number of woman entrants, persons	0.318	0.004	1.0	0.0
Number of man entrants, persons	0.353	0.0	1.0	0.06
Number of state order training woman entrants	0.586	0.0	0.837	0.08
Number of state order training man entrants	0.270	0.904*	0.953	0.197*
Number of paid training woman entrants	0.264	0.006	0.0	0.001
Number of paid training man entrants	0.147	0.0	0.304	0.780*
Number of woman students	0.746	0.001	0.840	0.144*
Number of man students	0.00	0.745	0.974	0.011
Number of state order training woman students	0.653	0.009	0.639	0.0
Number of state order training man students	0.710	0.86*	0.001	0.011
Number of paid training woman students	0.991	0.0	0.516	0.323*
Number of paid training man students	0.390	0.077	0.505	0.002
Number of woman graduates	0.997	0.154*	0.0	0.0
Number of man graduates	0.565	0.0	0.683	0.530*
Number of foreign woman entrants	0.195	0.0	0.417	0.001
Number of foreign man entrants	0.640	0.0	0.0	0.25*
Number of foreign woman students	0.382	0.53*	0.908	0.033
Number of foreign man students	0.502	0.0	0.191	0.0

Asterisk (*) denotes a non-stationary time series, which is removed to obtain our final set of series for both experiments.

Table 2: Masters Experiment Results.

Cluster Number	Euclidean			DTW			Soft DTW		
	Original	First-Order Differenced	Final	Original	First-Order Differenced	Final	Original	First-Order Differenced	Final
2	0.24	0.26	0.22	0.16	0.33	0.22	0.34	0.67	0.5
3	0.17	0.29	0.27	0.09	0.19	0.16	0.35	0.58	0.44
4	0.1	0.32	0.34	0.14	0.17	0.26	0.38	0.55	0.654
5	0.11	0.28	0.24	0.14	0.17	0.25	0.36	0.46	0.443
6	0.16	0.24	0.19	0.11	0.13	0.15	0.33	0.52	0.726
7	0.244	0.35	0.06	0.09	0.13	0.08	0.29	0.2	0.59
8	0.212	0.32	0.1	0.09	0.13	0.1	0.09	0.44	0.59
9	0.227	0.26		0.1	0.1		0.01	0.39	
10	0.214	0.25		0.1	0.07		0.29	0.4	

Table 3: PhD Experiment Results.

Cluster Number	Euclidean			DTW			Soft DTW		
	Original	First-Order Differenced	Final	Original	First-Order Differenced	Final	Original	First-Order Differenced	Final
2	0.35	0.26	0.244	0.33	0.34	0.34	0.65	0.65	0.63
3	0.36	0.26	0.21	0.18	0.19	0.31	0.61	0.5	0.49
4	0.29	0.23	0.20	0.19	0.18	0.21	0.53	0.6	0.19
5	0.27	0.22	0.22	0.24	0.17	0.16	0.47	0.49	0.09
6	0.30	0.20	0.20	0.25	0.14	0.10	0.46	0.22	-0.07
7	0.26	0.14	0.09	0.24	0.12	0.3	0.34	0.36	0.2
8	0.26	0.13	0.11	0.21	0.14	0.04	0.20	0.60	0.04
9	0.27	0.12		0.17	0.15		0.11	0.27	
10	0.29	0.14		0.17	0.11		0.11	0.11	

Table 4: PCA Results with Soft DTW.

Cluster Number	Masters Experiment			PhD Experiment		
	Original n=10	First-Order Differenced n=10	Final n=6	Original n=10	First-Order Differenced n=10	Final n=6
2	0.63	0.55	0.36	0.43	0.57	0.56
3	0.30	0.37	0.21	0.68	0.47	0.57
4	0.27	0.30	0.58	0.58	0.34	0.35
5	0.37	0.29	0.24	0.49	0.32	0.07
6	0.35	0.27	0.31	0.42	0.27	0.06
7	0.06	0.30	-0.03	0.32	0.05	0.26
8	0.07	0.37	0.22	0.38	0.16	0.20
9	0.24	0.10		0.39	0.40	
10	0.09	0.06		0.26	0.35	

Table 5: Average correlation per Dataset.

Dataset	Average Correlation
Masters	0.913
Masters Differenced	0.379
PhD	0.243
PhD Differenced	0.245

To be able to understand how much of this is due to limiting the number of features and what role stationarity plays in such results, we conduct a dimensionality reduction using PCA and observe the clustering results with the *Soft DTW* metric, which overall provides the best results among three distance metrics for each experiment. We select the maximum number of components to optimally describe the data. For both experiments, the number of components is 10 for the original and first-order differenced datasets, and 6 for the final ones, and this is a further indication of the overall similarity of the experiments. For the Masters experiment (see Table 4), PCA significantly improves performance over the original dataset, while reduces performance over the first-order differenced and final datasets. This finding demonstrates three points. The first one is that we only gain information while conducting a dimensionality reduction on a highly correlated non-stationary dataset, while loose information on partially and fully stationary datasets, even with the optimal number of clusters. Secondly, our first order differenced and final datasets from Table 2 contain

more information. This may also indicate that the drop in the performance of the final stationary dataset in Table 2 is mostly due to an information loss after dropping stationary time series features with a union operation, which removed stationary columns such as *Number of paid training man entrants, Number of paid training man entrants, Number of woman students, Number of paid training woman students, Number of man graduates, Number of foreign man entrants*. From the PhD experiment results in Table 3, we observe a performance drop in all the original, differenced and final datasets, unlike the Masters Experiment results, where the performance improved over the original dataset. Our findings allow us to summarize the results in a few key points.

- For highly correlated non-stationary datasets (e.g., *Masters Experiment*), irrespective of the distance metric, applying first-order differencing for making the dataset stationary improves the clustering results. Furthermore, our PCA findings show that for such datasets, applying clustering on the original dataset will hurt the performance and much better results can be obtained with a reduced number of features. For partially stationary and stationary datasets, PCA is unable to improve performance, and performance deterioration is observed. PCA improves the performance of a non-stationary dataset, which, however, is not as good as the one achieved with differencing. Whether dimensionality reduction can fully substitute the stationarity procedure needs to be further explored.
- For low correlated non-stationary datasets (e.g., *PhD Experiment*), making the datasets stationary has a nonsignificant impact on the performance improvement. We attribute this to the fact that in such cases, differencing further removes variable cointengration and may hurt the results. we also observe that good clustering in terms of the evaluation score may be achieved with nonstationary series as well. In such cases, PCA hurts the results no matter the series are non-stationary, partially, or fully stationary.

6. Conclusion

This paper explored stationarity, a condition whose effect and significance on the time series clustering performance has been little investigated. We show that for non-stationary datasets with high correlation, stationarity should be applied as a standard clustering procedure as it helps to improve clustering performance. Further insights are obtained with dimensionality reduction, which shows that in such cases the performance can be significantly improved over the original dataset, indicating that applying clustering on raw time series datasets may lead to a reduced performance. In such cases, PCA improves results of non-stationary datasets, the impact of which is still less than the clustering results of a partially stationary dataset achieved with first-order differencing. We also conclude that with lowcorrelated nonstationary datasets, achieving stationarity with differencing may hurt cointegration and further worsen the results, so one may need to particularly handle non-stationarity if the variables are highly correlated. To observe the full effect of non-stationarity, as part of our future work we may need to conduct experiments with multiple little non-stationary low/highly correlated and highly non-stationary low/highly correlated synthetic datasets.

REFERENCES

1. *Aghabozorgi S., Seyed Shirkhorshidi A. and Ying Wah T.* (2015), Time-series clustering – a decade review. *Information systems*, 53, PP.16–38. Available from: https: //doi.org/https://doi.org/10.1016/j.is.2015.04.007.
2. *Chis M., Banerjee S. and Hassanien A.E.* (2009). Clustering time series data: An evolutionary approach, vol. 206. PP. 193–207. Available from: https://doi.org/10.1007/ 978-3-642-01091-09.
3. *Cuturi M. and Blondel M.* (2017). Soft-dtw: a differentiable loss function for time-series. *International conference on machine learning*. PMLR. PP. 894–903.

4. *Javed A., Lee B.S. and Rizzo D.M.* (2020). A benchmark study on time series clustering. arxiv preprint arxiv: 2004. 09546.
5. *Keogh E.J. and Pazzani M.J.* (2000). A simple dimensionality reduction technique for fast similarity search in large time series databases. In: T. Terano, H. Liu and A.L.P. Chen, eds. Knowledge discovery and data mining. current issues and new applications. Berlin, Heidelberg: Springer Berlin Heidelberg. PP. 122–133.
6. *Lin J., Keogh E., Lonardi S. and Chiu B.* (2003). A symbolic representation of time series, with implications for streaming algorithms. Proceedings of the 8th acm sigmod workshop on research issues in data mining and knowledge discovery. PP. 2–11.
7. *Makhlouk O.* (2018). Time series data analytics: clusteringbased anomaly detection techniques for quality control in semiconductor manufacturing. Ph.D. thesis. Massachusetts Institute of Technology.
8. *Mohamad I.B. and Usman D.* (2013). Standardization and its effects on k-means clustering algorithm. Research journal of applied sciences, engineering and technology, 6 (17). PP. 3299–3303.
9. *Piech C.* (2013). K means: https://stanford.edu/cpiech/cs221/handouts/kmeans.html.
10. *Potikyan N.* (2020). Clustering financial time series. Master's thesis. University of Tartu.
11. *Sambandam R.* (2003). Cluster analysis gets complicated. Marketing research, 15. PP. 16–21.
12. *Shalizi C.* (2012). Principal Components Analysis.
13. *Wang X., Smith K.A., Hyndman R. and Alahakoon D.* (2004). A scalable method for time series clustering. Unrefereed research papers, 1.
14. *Xiong Y. and Yeung D.Y.* (2002). Mixtures of arma models for model-based time series clustering. 2002 ieee international conference on data mining, 2002. proceedings. IEEE. PP. 717–720.

ВЛИЯНИЕ СТАЦИОНАРНОСТИ НА КЛАСТЕРИЗАЦИЮ ВРЕМЕННЫХ РЯДОВ

К.М. Гишян, Х.С. Смбатян

Университет Бат (Великобритания),
Ереванский Государственный Университет

АННОТАЦИЯ

В данной научной статье речь идет о том, что большинство существующих приложений кластеризации временных рядов полагаются на выполнение кластеризации исходных или необработанных данных. В этом контексте мы обнаруживаем большой пробел в литературе, поскольку такой подход игнорирует стационарность, которая предполагает, что среднее значение, дисперсия и автокорреляция ряда остаются постоянными во времени. Для двух наборов данных с наблюдениями временных рядов, которые очень нестационарны и позже преобразуются, чтобы стать частично стационарными и стационарными, мы исследуем производительность алгоритма кластеризации с тремя показателями расстояния, а затем выполняем уменьшение размерности с помощью PCA для дальнейшей оценки результатов. Мы пришли к выводу, что, хотя нестационарность не может ухудшить производительность кластеризации, выполнение стационарного набора данных для высококоррелированных нестационарных наборов данных может значительно улучшить производительность кластеризации и должно быть частью стандартной процедуры кластеризации временных рядов. Дальнейшая работа может включать изучение производительности нескольких наборов данных с разной степенью мультиколлинеарности и стационарности для более полного понимания эффекта

Ключевые слова: временные ряды, кластеризация, стационарность.

УДК004.05

Поступила: 13.04.2021 г.
Сдана на рецензию: 14.04.2021г.
Подписана к печати: 22.04.2021г.

DESIGN AND DEVELOP METHODS FOR FUZZ TESTING IMPROVEMENT

J. Hakobyan

Yerevan State University

jivan@ispras.ru

ABSTRACT

In this paper, we present a novel extendable fuzzing framework. The framework can be extended to include new functionalities, which improve testing results. In order to achieve extendibility, the framework supports a flexible plugin management mechanism. All features of our fuzzing tool are added as plugins and controlled via plugin manager. The framework supports a number of default plugins for different tasks, such as input data mutations, internet of things (IoT) devices fuzzing, network protocol testing, dynamic symbolic execution invocation, target program run, coverage collection, etc. The tool is capable of performing fuzzing for different inputs: files, standard input, network, command-line arguments, and environment variables.

During the experimental setup, the tool found more than 280 defects on 14 programs under different Linux distributions. Also, it has found 15 unique crashes on the Samsung Smart Things application, which were approved by Samsung Electronics R&D.

Keywords: fuzzing, IoT, mutation.

1. Introduction

Fuzzing [1] is one of the most used and efficient dynamic analysis methods. It finds various types of defects through the execution of the target

program. Fuzzing tools generate test cases and pass them to the target program as input data. They usually repeat the following steps: 1) generate input data, 2) execute target program with generated data, and 3) follow execution process to record crashes/hangs. The results of testing are closely dependent on generated input data.

AFL [2] is a state-of-the-art fuzzing tool. It can test programs, which receive input data through a file or standard input. AFL collects code coverage and uses a genetic algorithm to improve the quality of generated data. It uses two types of instrumentation for code coverage collection: dynamic and static. Static instrumentation is performed during program compilation. Modified compiler referred to as AFL-GCC injects special instructions for coverage collection to binary code basic blocks (BB). For dynamic instrumentation, AFL uses a modified version of Qemu [3]. In the latter, the program's source code is not required. Lib Fuzzer [4] is intended for the testing of separate functions. The user should write extra code from which the target function is called. Peach [5] is a protocol fuzzing tool, which uses special descriptions for target protocol.

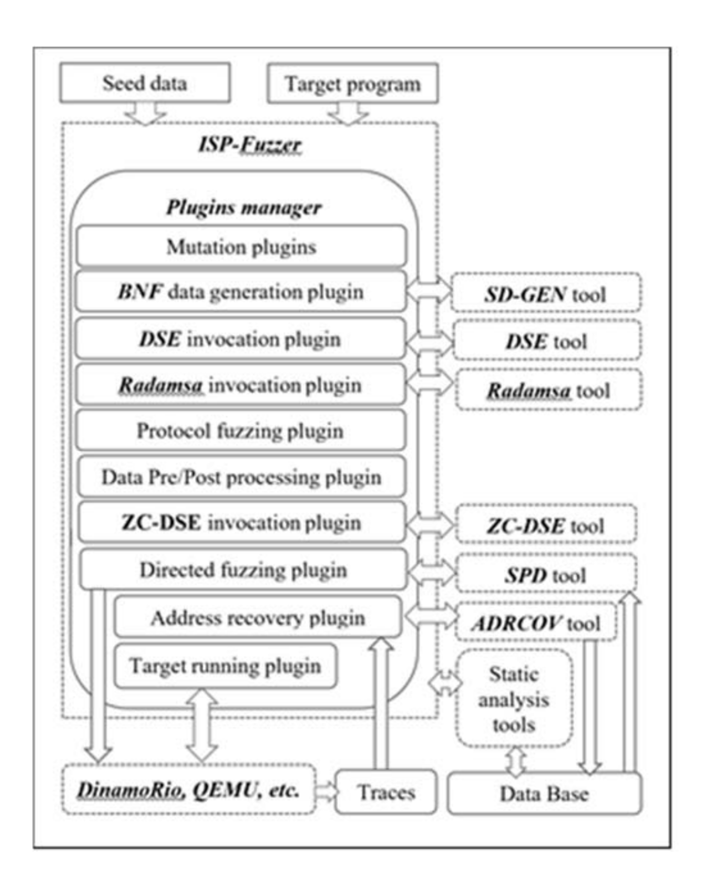
Special attention should be paid to the library function testing. Besides testing each function separately, an effective fuzzing tool should test a combination of library functions (chains of library functions call). Library function combination testing may reveal defects, which will not be detected in the case of a separate testing. To the best of our knowledge, for the testing of IoT [6] software there is no fuzzing tool, which can construct functions call chains, where the return value of one function is used as an argument in the other.

Existing fuzz testing tools cannot fully test software that receives input in several ways. In addition, no testing tool will take into account the chain of the library function calls. Therefore, we design and develop a fuzz testing framework which is applicable to Linux and Windows operating systems. It contains a flexible and extendable plugins management engine.

2. Proposed fuzzing tool overview

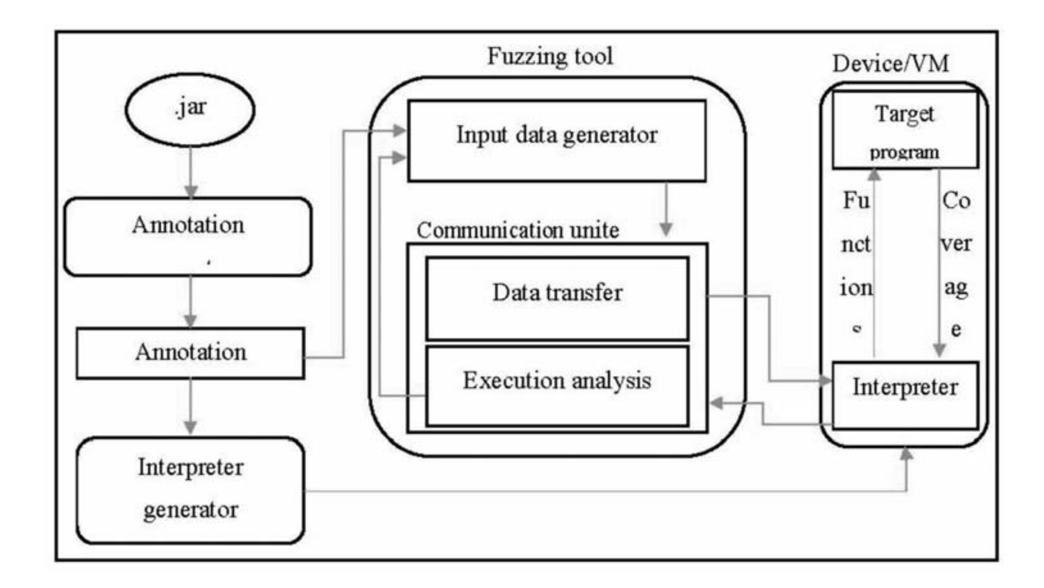
2.1. Fuzzing framework description

To make the proposed fuzzing framework easily adaptable for different tasks we have developed a plugins mechanism (picture 1). All basic components of the framework are released as plugins. Users can add custom functionality if existing plugins do not fit the solution of the problem. All plugins have direct access to program execution trace and DB, which keeps internal representation of target binary. More information about implemented plugins can be found in [7] paper.



Picture 1. Fuzzing tool overview.

The core of the framework implemented in C/C++ as a platform-independent unit. For parallel and distributed execution, we have implemented a separate tool. The tool is responsible for the run and management of fuzz instances. It only requires a shared folder between distributed machines. Each fuzzing instance creates its working directory on that shared folder and keeps execution results there. Periodically instances synchronize with each other (import new discoveries from other instances). Users can interactively increase or decrease fuzzing instances count.



Picture 2. API fuzzing workflow.

2.2. API fuzzing description

API fuzzing is fully automated and does not require any manual intervention. For that purpose we added three separate plugins to ours. The first plugin gets a jar file as input and generates annotations for its functions and classes. Annotations contain all necessary information about the classes

and functions (names, arguments list for each function).Input generator uses annotations to generate a valid chain of function calls with corresponding arguments. The second plugin uses annotations to generate functions calls chain interpreters. The interpreter works on the target environment and performs API functions calls. The interpreter gets functions calls chain, iterates through them and makes calls. During function calls, the interpreter follows the status of execution. If a defect is detected then the interpreter interrupts execution and sends a message about the defect. After the execution of the whole functions calls chain, the interpreter collects code coverage and sends it back to the fuzzing tool. The last plugin is intended for communication between the fuzzing framework and target environment (it may be a separate device or virtual machine), where target library functions are executed. The plugin sends generated functions calls chain and receives execution status and code coverage.

Functions call chain generator:1) randomly selects functions from the library as initial call chain,2) iterates through arguments list of initial chain functions, 3) for each function argument finds other function from the library that returns a value of the same type,4) inserts calls of detected functions into initial call chain, then uses return values as arguments for corresponding functions,5) if no function is detected for given argument, corresponding object constructor is called to generate required value. Users can manage the length of the call chin through a configuration file. The platform keeps all interesting chains and does mutations on them. More information about the platform can be found in [8] paper.

Usually, IoT systems are made as libraries and have public methods for communication. The main goal of our tuned framework is testing IoT systems.

3. Results

We have evaluated the developed framework on different binaries from different OS. Found more than 280 defects on 14 programs under different Linux distributions (Table 1). All results are manually verified.

Table 1.

Results of experimental testing of ISP-Fuzzer

Name	OS	Plugin	Error type	Number of defects
colcrt	Ubuntu-18.04	-	Crash	19
column	Ubuntu-18.04	-	Crash	7
gtbl	Ubuntu-18.04	-	Crash	20
troff	Ubuntu-18.04	-	Crash	4
tbl	Ubuntu-18.04	-	Crash	24
exiv2	Ubuntu-16.04	-	Hang	1
captoinfo	Ubuntu-16.04	-	Crash	1
gtbl	Ubunut-16.04	-	Crash	20
infotocap	Ubuntu-16.04	-	Crash	1
exiv2	Ubuntu-16.04	DSE	Crash	1
bison	Ubuntu-16.04	-	Hang	2
jasper	Ubuntu-16.04	DSE	Crash	19
colcrt	Ubuntu-16.04	DSE	Crash	1
iptables-xml	Ubuntu-16.04	DSE	Crash	2
dc	Ubuntu-16.04	DSE	Crash	4
teckit_compile	Ubuntu-16.04	DSE	Crash	1
tic	Debian 6.0.10	DSE	Crash	2
tbl	Debian 6.0.10	DSE	Crash	153

Besides the mentioned defects, IoTAPI fuzzing revealed15 unique crashes in the Samsung Smart Things [9] application (Plugin Base API library). All of them were approved by Samsung Electronics R&D.

4. Discussion

In this paper, we presented anew extendable and efficient fuzzing platform. It can easily be adapted to new fuzzing tasks. API fuzzing

approach proved its efficiency based on obtained results for real-world IoT applications. Experimental results have shown that the platform can efficiently detect defects in real-world software.

REFERANCES

1. Fuzzing (online publication). Available at: https://en.wikipedia.org/wiki/Fuzzing, 09.04.2021
2. American fuzzy lop (online publication). Available at: http://lcamtuf.coredump.cx/afl 11.12.2018
3. Qemu (online publication). Available at: https://wiki.qemu.org/Main_Page
4. libFuzzer. Available at: https://llvm.org/docs/LibFuzzer.html
5. Peach (online publication). Available at: https://www.peach.tech/products/peach-fuzzer, 01.07.2019
6. Internet of things (IoT) (online publication). Available at: https://en.wikipedia.org/wiki/Internet_of_things
7. ISP-Fuzzer: Extendable fuzzing framework S. Sargsyan, J. Hakobyan, M. Mehrabyan, M. Mishechkin, V. Akozin, Sh. Kurmangaleev, Proceedings of Ivannikov Memorial Workshop, Velikiy Novgorod, Sep 2019, PP. 68–71, DOI: 10.1109/IVMEM.2019.0001
8. Automatic API fuzzing framework Sargsyan S.S., Vardanyan V.G., Hakobyan J.A., Aghabalyan A.M., Mehrabyan M.S., KurmangaleevSh.F., GerasimovA.Yu., Ermakov M.K., Vartanov S.P. Proceedings of the Institute for System Programming, vol. 32, issue 2, 2020. PP. 161–174 DOI: 10.15514/ ISPRAS-2020-32(2)-13.
9. Smart Things, Available at: https://www.samsung.com/global/galaxy/apps/smartthings/.

ИССЛЕДОВАНИЕ И РАЗРАБОТКА МЕТОДОВ НАПРАВЛЕННЫХ НА ПОВЫШЕНИЕ ЭФФЕКТИВНОСТИ ФАЗЗИНГА

Дж. А. Акопян

Ереванский Государственный Университет

АННОТАЦИЯ

В данной научной статье представляется новая и развивающаяся среда фазз-тестирования. Данная среда может развиваться и включать новые функциональные возможности для улучшения результатов тестирования. Среда состоит из плагинов, которые возможно изменять или добавлять, используя менеджер плагинов. Фреймворк содержит ряд реализованных плагинов, которые обеспе-чивают широкий набор функционала: мутации данных, фаззинг интернет вещей (IoT), тестирование сетевых протоколов, интеграцию динамического символь-ного исполнения, запуск целевой программы и т.д. Среда может выполнять фаз-зинг для файла, стандартного ввода, сети, аргументов командной строки, пере-менных окружения.

Во время экспериментального запуска данный инструмент выявил более 280 дефектов в 14 программах в разных дистрибутивах Linux. Кроме того, были выявлены 15 уникальных сбоев в приложении Samsung Smart Things. Все они подтверждены отделом исследований и разработок Samsung Electronics.

Ключевые слова: фаззинг, интернет вещей, мутация.

Удк 004.855.5

Поступила: 16.04.2021г.
Сдана на рецензию: 20.04.2021г.
Подписана к печати: 29.04.2021г.

DETECTION AND PREDICTION OF PUMP AND DUMP EVENTS USING MACHINE LEARNING MODELS

H. Zohrabyan, G. Grigoryan

Yerevan State University
Shanghai University of Economics and Finance

hovhannes.zohrabyan@yahoo.com, gev_grigoryan@yahoo.com

ABSTRACT

The growing popularity of cryptocurrencies attracts increasingly more amount of people into the market. Such popularity and increasing number of newcomers into the market, creates a fertile field for fraudsters to cheat on public. One of the popular ways of fraud is the new formation of conservative pump-and-dump schemes. By the time, pump-and-dump schemes attracted the attention of cryptocurrency regulators and exchanges trying to predict pump-and-dump events beforehand in order to make exchanges more secure for the general user base. Such an algorithm can help them prevent pump-and-dump events from happening by limiting buy and sell volumes of the coin predicted to be pumped. As a result, cryptocurrency trading can potentially become more secure. Other works in this field have information about applying machine learning models trying to predict upcoming pumps.

Keywords: Machine Learning, Classification problems, Financial analysis, Crypto market, Pump and dump events.

Introduction

Traditional pump-and-dump manipulation schemes are based on the sequence of actions where the perpetrator first acquires a large long

position, then publishes false information to induce market participants to push prices up by buying the asset, and finally liquidates his own long position at a profit [1]. Pump-and-dump schemes are used for a long time in the conventional stock markets[8], however, as cryptocurrencies and modern communication technologies such as Telegram allow for more anonymity, these schemes started to occur on crypto exchanges as well [11].

In crypto exchanges pump-and-dump events mostly occur on coins with low market capital as shown later in the research. We will analyze features of the market and patterns of pump-and-dump events in order to train a machine learning model predicting them with high accuracy [7]. Some of the remarkable works in the field are researches conducted by Jiahua Xu [5] and Kamps, Josh and Kleinberg [10]. We used the research mentioned above to compare the results of different algorithms with Jiahua Xu's results to test new algorithms that may behave better with higher accuracy.

1. Background to pump-and-dump events

Pump-and-dump schemes are based on the sequence of coordinated actions, which brings to a short-term increase of price resulted from high demand in the exchange [5]. These actions are coordinated by the perpetrator/admin, who first acquires coins for himself, then publishes information to other participants to start acquiring coins, which results in high demand, therefore higher prices. After the prices spike, the perpetrator/admin sells the acquired amount of coins in order to get profit.

1.1. Described process of pump-and-dump event

Pump-and-Dump event is organized by individuals or groups. Pump organizers coordinate actions of people who buy and sell crypto coins. People participating in the pump-and-dump event follow the signals of pump organizers to start buying a particular coin. Admins broadcast a message about the upcoming pump beforehand, giving the exact time and

name of the exchange where the pump should happen. A few seconds before the announced time, the name of the coin is published, usually in a form that is hard to recognize by computer vision algorithms. Immediately after the pump started, the price of the coin inflates, sometimes increasing several folds. Eventually, the first price drop will appear, and members start to sell coins as fast as possible, which results in the deflation of price, and usually, price drops lower than the before-pump price. Such schemes are most profitable for the administrators of the groups, as they buy coins beforehand with low prices and sell them with inflated prices during the event getting great profit.

2. Depth review of Pump-and-Dump schemes

Even though Pump-and-Dump events differ from each other by the pump amount, sell and buy volumes, etc., they all have a common pattern that can be detected to be implemented into the algorithm. To understand those patterns, we will analyze multiple real pump-and-dump cases in crypto exchanges.

We chose four pump events that happened on the "Binance" to work with. The pump-and-dump events were targeting coins named WINGS, VIA, STORJ and GXS, and happened in the timeframe from December 15, 2018, to February 20, 2019.

2.1. Phases of Pump-and-Dump event

Before analyzing the features and patterns of the pump-and-dump events, we will break down such events into three diverse and abstract phases. As mentioned in section 1.1, pump administrators acquire coins beforehand at a lower price to maximize their profit. However, as the coins that pump-and-dump targets have low market capital, as a result, the act of acquiring coins by administrators also affects the price. We are going to name this timeframe the "pre-pump phase". This phase will be the main target for the ML model. The timeframe after the start of the pump event

and before the first price drop will be called “pump phase”. And lastly, the timeframe after the first price drop will be called “dump phase” [12].

2.2. Price change patterns

The four graphs (Figure 2, 3, 4, 5) presented below illustrate the price change, accompanied by volumes of buy and sell starting from one hour before the pump and up to 4 hours afterward. These graphs perfectly show the spike that happens during the pump. On these graphs, we can observe normal prices and a relatively flat line before the pump start, described as the “pre-pump phase”. After the first few seconds, the volumes of buy orders increase significantly. The increased volumes of buy orders result in immediate inflation of the price in the “pump phase”. After the first price drop, the buy volumes decrease while sell orders increase, which brings the prices back to the price of the coin before the pump as is typical to “dump phase”.



Figure 2. Price plot of WABI.



Figure 3. Price plot of GXS.



Figure 4. Price plot of VIA.

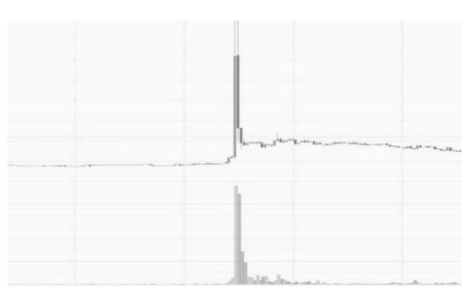


Figure 5. Price plot of STORJ.

2.3. Volume change patterns

As it has already been stated, during pump events, the price of coin inflates, resulting from the increased volume of people trying to buy the same coin at the same time. The graphs represented in Figure 6 show how the pump phase is accompanied with increasing volumes of buy orders, which is natural to observe. Moreover, all four graphs show the increasing amount of sell orders as soon as the prices started to drop, which was described in section 2.1 as the "dump phase". However, on the first graph, representing the pump event of the GXS coin, we can see that after the pump, in the dump, phase volumes started to increase one more time.

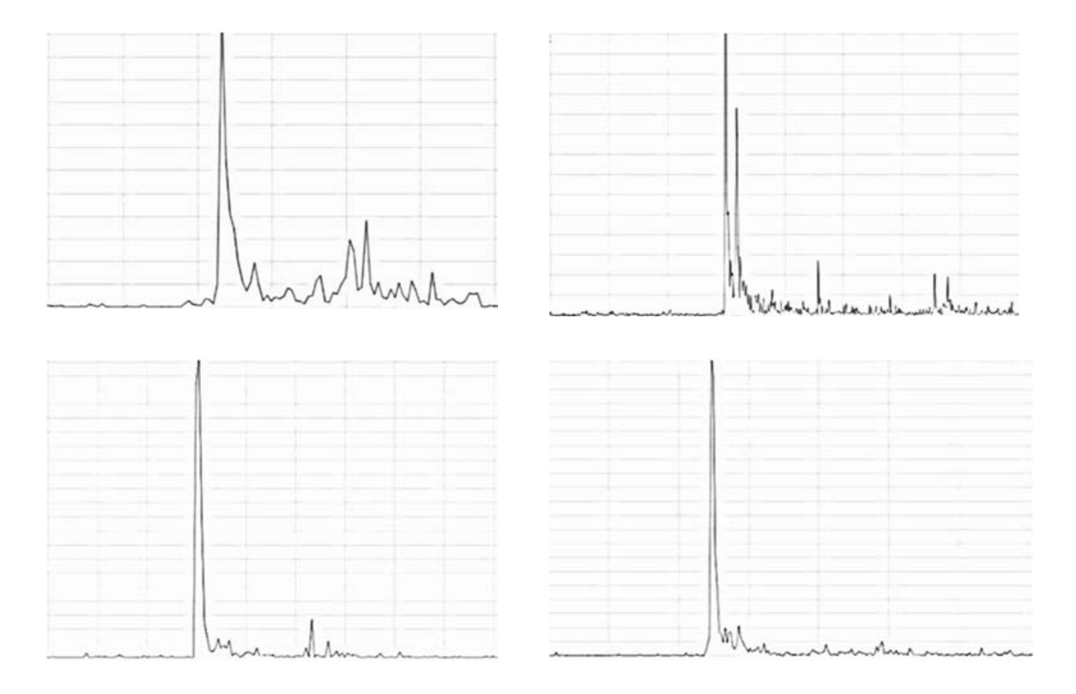


Figure 6. Volume Plots of WABI, GXS, VIA and STORJ during corresponding pump events.

3. Analyzing Pump-and-Dump Data

In order to be able to analyze data and calculate features that are most efficient for a machine learning model, data has to be gathered and cleaned. First of all, hourly historical data of coins has to be collected. Afterward, data about past pump-and-dump events has to be collected and combined with actual historical data.

3.1. Collecting Historical Data

To collect hourly data of coins, we will need to fix three important variables: the set of exchanges, the set of coins, and pairs. For this research, four exchanges are chosen to be observed in addition to global data are: “Binance”, “Yobit”, “Bittrex” and “Cryptopia”. A set of 577 unique coins will be fixed to collect from all above-mentioned exchanges paired with BTC.

The public API of “Crypto Compare” is used to collect data, giving access to hourly OHLC (open, high, low, close) and volume data of given pair in the selected exchange. Another important parameter is the Market Capital of the coin. In the conventional stock market, it is expected for pump-and-dump events to be orchestrated on stock with low market capital [7]. The pattern has to be the same for the crypto market as well, as lower market capital allows easier manipulation of price [5]. For this matter, we will use API from “Coin Market Cap” to gain the market capital of all coins at a fixed timeframe of December 1, 2019 [5]. We believe that other features of coins are not important enough to affect the results of the machine learning algorithm, thus we do not collect them.

3.2. Collecting and Combining Pump-and-Dump data

After the historical data of coins is available, the corresponding data of pump-and-dump events has to be collected. “Pump Olymp” provides a list of ended pumps for the needed time interval. The data includes the name

of the coin, exchange, timestamp, estimated profit, etc, which is more than enough to analyze and combine with actual historical data.

Given the historical OHLC data for the coins and exchanges, the data has to be combined with pump-and-dump events. For this purpose, we need to find the best fitting timestamp in historical data for each timestamp of the pump-and-dump event. This can be accomplished by calculating the minimal difference between all possible combinations of data points. However, it is important to be attentive for the difference to be less than 3600 seconds, as most pumps do not take more than an hour, and we do not want to label data without the pump happening as a "Pump". As it can be observed from the graph in Figure 7 most points, do not differ from each other for more than a half-hour, and others are exactly the same, which is an excellent result. This means that the data can be combined without significant information loss.

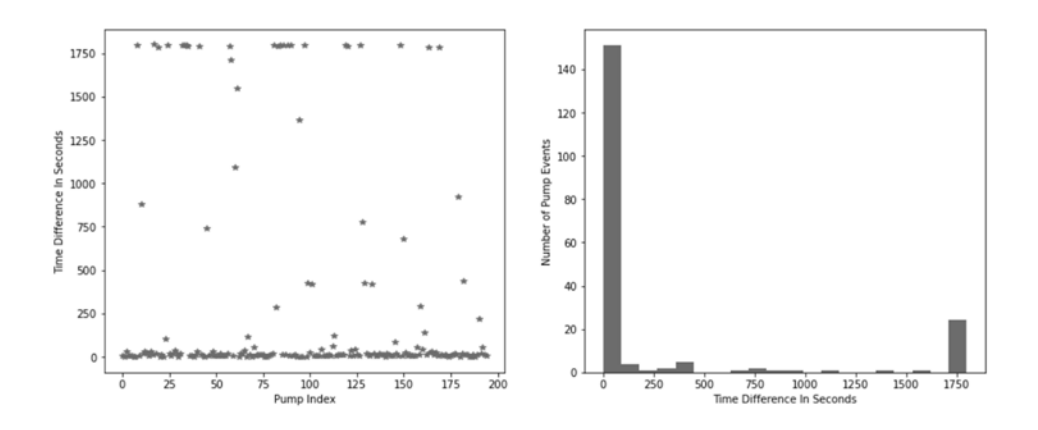


Figure 7. Difference of historical data and pump data timestamps.

3.3. Importance of the Exchange

Pump-and-Dump events have to happen in one of the exchanges available."Binance" and "Bittrex" are less popular as they mainly host coins

with large market capitals. In comparison, “Cryptopia” and “Yobit” host coins with lower market capitals which are easier to manipulate, resulting in higher popularity among pumping groups. This suggestion can be confirmed by pump-and-dump data in each exchange seen below (Figure 8). Based on this data, we will focus on pump-and-dump events that happened in “Cryptopia”.

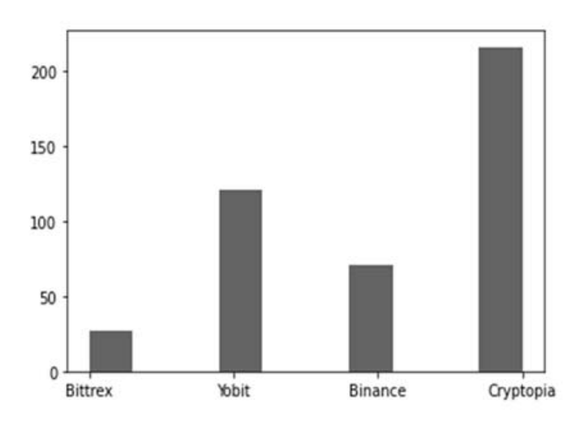


Figure 8. Distribution pump-and-dump events between the exchanges.

4. Predicting Pump-and-Dump Events

4.1. Choosing Features for the Machine Learning model

As the paper has already shown, pump-and-dump events have patterns that repeat. Thus, theoretically, there is a possibility to use market indicators and features based on them to predict upcoming pump-and-dump events. When combining historical data and pump-and-dump event data, we will mark each point with a Boolean value representing either being a pump event or not. Moreover, as we have already seen, pump organizers start buying coins prior to the pump to maximize their profit. As a result, we can base our features on time windows of different lengths, which will provide us with data of coin state before the pre-pump phase and state during the

pre-pump phase. In order to achieve such results, timeframes of lengths 3, 12, 24, 36, 48, 60, and 72 hours are chosen.
Given the timeframes that will be used, features have to be chosen to be calculated on them. One of the most common descriptors of the trend of the market is the price return [7]. In this paper, an x hour return will be calculated with 1-hour margin from the pump signal

$$r_x = \frac{P_{t-1} - P_{(t-1)-x}}{P_{(t-1)-x}}$$

where x ∈ [3, 12, 24, 36, 48, 60, 72] and *t* is the time of the pump signal.

We also want to know how much the one-hour return fluctuated [7] in the given time frame. For that reason, the variance of one-hour returns will be calculated, which is also known as return volatility

$$returnvolatility_n = Var(r_1^n)$$

where n ∈ [3, 12, 24, 36, 48, 60, 72] and r_1^n is the set of one hour returns in the time frame t-1 to (t-1)-n.

Given the returns and return volatility, the algorithm can track the price changes. However, we have seen in the analysis that buy and sell volumes change as well and have small fluctuation in the pre-pump phase and great changes during the pump phase. As a result, the volume-from and volume-to and their volatility can be considered as features. Volume-from and volume-to represent data points from the collected dataset at the given timestamp of t–x and defined as vx. The volatility of volume-from and volume-to is defined as follows

$$volumevolatility_n = Var(v_n)$$

where n ∈ [3, 12, 24, 36, 48, 60, 72] and v_n is the set of volume-from/volume-to for the time frame t-1 to (t-1)-n.

In addition to these parameters, last hour price, return, volume-from, and volume-to will be added as features. Through experiments on the machine learning models described later, the market capital was classified redundant (not affecting the accuracy of the model).

4.2. Creating Training Dataset

In order to create data to train machine learning models on, historical data and pump data have to be used to create a new dataset with the features mentioned above. A python script will find the closest historical data point for each pump case, retrieved from pump dataset. After that, for every coin fixed, the above-described features will be calculated, getting the label "pumped = False," and only the coin mentioned in pump data will get the label "pump = True". As a result, for each pump case in the pump dataset, 576 "pump = False" labeled data points will be added.In the pump dataset, approximately 192 pump cases were retrieved. The above algorithm results in data consisting of 192 pumps and 110592 non-pump datapoints. Such data can be considered extremely skewed and requires subtle handling in the models. Finally, with all features calculated, the dataset will have the columns illustrated in Figure 9.

```
'Name', 'Symbol', 'datetime', 'pumped', 'last_price', 'last_volumefrom',
'last_volumeto', 'last_return', 'return3h', 'return12h', 'return24h',
'return36h', 'return48h', 'return60h', 'return72h', 'returnvola3h',
'returnvola12h', 'returnvola24h', 'returnvola36h', 'returnvola48h',
'returnvola60h', 'returnvola72h', 'volumefrom3h', 'volumefrom12h',
'volumefrom24h', 'volumefrom36h', 'volumefrom48h', 'volumefrom60h',
'volumefrom72h', 'volumefromvola3h', 'volumefromvola12h',
'volumefromvola24h', 'volumefromvola36h', 'volumefromvola48h',
'volumefromvola60h', 'volumefromvola72h', 'volumeto3h', 'volumeto12h',
'volumeto24h', 'volumeto36h', 'volumeto48h', 'volumeto60h',
'volumeto72h', 'volumetovola3h', 'volumetovola12h', 'volumetovola24h',
'volumetovola36h', 'volumetovola48h', 'volumetovola60h',
'volumetovola72h'
```

Figure 9. List of columns of training dataset.

4.3. Handling Imbalanced Datasets

The imbalance of the dataset affects the training and testing of the machine learning model. Both training and testing datasets have to be chosen carefully to get real and precise measurements of the model's performance. There are two popular ways to model evaluation train-test split and k-fold cross-validation. Both of these algorithms are excellent when working with balanced data but can result in inaccurate results in case of imbalanced data [6]. In the case of the classification model, evaluation of the accuracy is more important and more complicated. In order to evaluate a classification model, a set of labeled data different from the train dataset is necessary. To acquire such data, developers use the train-test split, where the dataset is cut into two parts with a certain percentage (50–50%, 80–20%). However, as already mentioned, both of these algorithms fail to give accurate results in case of highly imbalanced datasets. The reason for the failure is that generated testing dataset will be imbalanced as well, and because it is only a sample from the big dataset there is a high probability of having only a few datapoints of the minor class. We will use the technique of changing the accuracy measurement. Instead of measuring the percentage of right prediction from whole data, one of the measures of “Confusion Matrix”, “Precision”, “Recall” and “F1 Score” can be used [6].

4.4. Selecting Models

To predict upcoming pumps, we will use different algorithms of classification and logit regression [17]. As it was already mentioned, the training dataset is heavily skewed, which results in bad performance for some models. We will use some of the best performing models used in classification problems, such as Adaptive Boost, Gradient Boosting, KNN, and Random Forest. We do not expect Adaptive Boost and Gradient Boosting to perform well as these algorithms are not suited for working well with skewed data [13]. On the other hand, the KNN algorithm can show good performance but needs Hyperparameter tuning to increase

performance [15]. Finally, the specifics of the Random Forest algorithm allowing to choose a random sample for training of each sub-tree allows to handle skewed data. The process of choosing random sub-samples or otherwise called stratification, allows us to give weights to classes in the sample, thus eliminating the skew of data. Moreover, this algorithm handles a big number of features remarkably, which in our case is 50. However, Random Forest is not intuitive in providing information about feature importance [16]. For the logit regression model, GLM or Generalized Linear Model will be used. In contrast to Random, Forest GLM performs poorly in the case of overfitting features (feeding too many features). However, GLM has a very optimized algorithm allowing it to handle big datasets with ease while still providing acceptable results. Finally, performance measures for all above mentioned models are available In Figure 11. For Random Forest and Adaptive Boosting Model the ROC Curves are illustrated in Figure 12.

4.5. Adaptive Boosting Model

In order to implement the ADA boost algorithm, we will use "Scikit-learn" [3] library's ensemble package [14] containing a great amount of different ensemble classification models [16]. For the ADA boost classifier hyperparameter, "n_estimators=100" will be defined. After setting up the train dataset and importing all needed modules, the ADA boost classifier can be trained. The classifier was trained on train data, which is an 80% split of calculated data. Other 20% was used for testing purposes.

We will be looking for three parameters: ROC curve, Precision, and F1 Score to analyze the model's performance. Figures 11 shows that precision for predicting pumps is 56%, with an f1 score of 0.21.

4.6. Gradient Boosting Model

Gradient Boosting Model will be trained in the same way as previous model. For this model,we will define Hyperparameters "n_estimators=100",

"learning_rate=1.0", "max_depth=10" However, as we can see from performance measures of Gradient Boosting, the model performed bad with low accuracy and f1-score for pump cases as it was expected.

4.7. GLM Model

For the GLM model, we will use the implementation of "Stats Models" [4]. We will train two generations of the GLM model, each getting different parameters, by purging the ones marked as non-important by the previous generation. For the first GLM, we will not remove any features. After that, we can choose a set of features to be dropped. For the second generation, we will keep only columns representing 1, 36 and 48-hour returns and print report to evaluate the performance (Figure 12).

4.8. K-Nearest Neighbors model

The KNN model is also implemented using the "Scikit-learn" model. The KNN requires three hyperparameters of leaf size, the number of neighbors, and the type of distance to use, either Euclidean or Manhattan. In order to find the best parameters for the model, we do hyperparameter tuning using GridSearch CV (implemented from "Scikit-learn"). We fixed possible parameters of leaf size in range (1, 30), the number of neighbors in (1, 10), and parameter "p" responsible for distance measurement algorithm, which can be either 1 or 2.The GridSearch results for train data showed optimal parameters of "leaf size = 30", "n_neighbors=7", and "p = 2". After training the KNN model with the following hyperparameters, we see precision for predicting pumps is 75%, with an f1 score of 0.86.

4.9. Random Forest Model

As it was already mentioned, Random Forest can use stratification in order to handle imbalanced data and created a balanced sample. In the python implementation, the stratification cannot be controlled precisely by

giving a number of elements from each class like in R. In this case, we will use the arguments "bootstrap = True" and "class weight = balanced subsample". These arguments allow the Random Forest classifier to stratify the main dataset by selecting entries from each class while keeping the balance in the subsample.

For this model, precision for predicting pumps is 92% with an f1 score of 0.70, which is an excellent result.

Model Name	"Not-Pump" Precision	"Pump" Precision	"Not-Pump" Recall	"Pump" Recall	"Not-Pump" f1-score	"Pump" f1-score	Accuracy
Adaptive Boosting	1.0	0.56	1.0	0.13	0.21	0.21	1.0
Gradient Boosting	0.99	0.04	0.17	0.95	0.29	0.08	0.29
K-Nearest Neighbors	1.0	0.75	0.99	1.0	0.99	0.86	0.99
Random Forest	0.98	0.92	1.0	0.56	0.99	0.78	0.98

Figure 10. Model measures report.

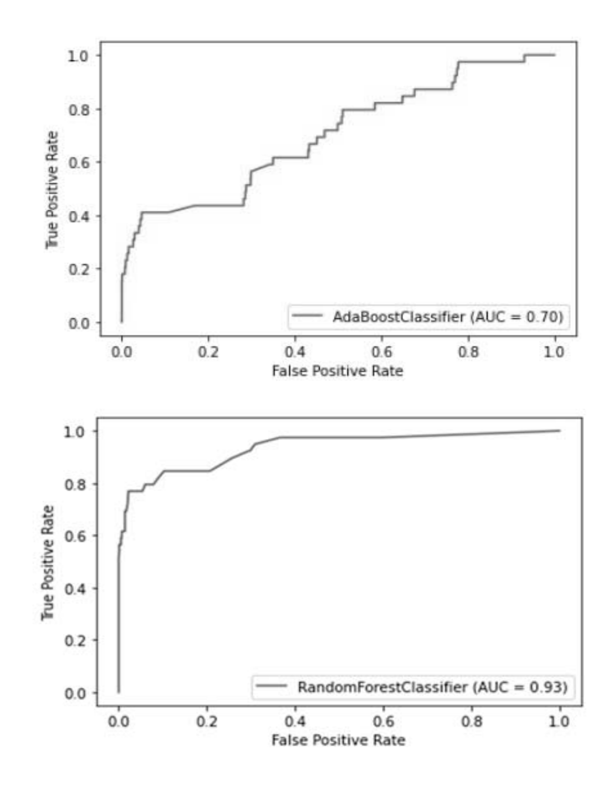


Figure 11. ROC Curves of AdaBoost and Random Forest Classifiers.

```
                 Generalized Linear Model Regression Results
==============================================================================
Dep. Variable:                 pumped   No. Observations:                47114
Model:                            GLM   Df Residuals:                    47113
Model Family:                Binomial   Df Model:                            0
Link Function:                  logit   Scale:                          1.0000
Method:                          IRLS   Log-Likelihood:                -517.91
Date:                Sat, 10 Apr 2021   Deviance:                       1035.8
Time:                        02:04:07   Pearson chi2:                 4.57e+04
No. Iterations:                    10
Covariance Type:            nonrobust
================================================================================
                  coef    std err          z      P>|z|      [0.025      0.975]
--------------------------------------------------------------------------------
datetime    -4.316e-09   8.29e-11    -52.047      0.000   -4.48e-09   -4.15e-09
last_return     4.4484      0.813      5.470      0.000       2.854       6.042
return36h       1.4267      1.100      1.297      0.195      -0.729       3.583
return48h       2.1819      1.080      2.020      0.043       0.064       4.299
================================================================================
```

Figure 12. GLM2 Results.

Conclusion

As a conclusion of the study conducted, we showed that it is possible to predict the upcoming pump-and-dump events using machine learning algorithms. Testing different models, while considering the fact that data is severely skewed, we found out that out of five models tested, Random Forest showed the best performance. Thanks to the stratification algorithm in Random Forest was able to get 92% accuracy in predicting pump events. For future research, an algorithm for handling skewed data, possibly including stratification, before feeding to the model can be developed. Such algorithm can allow usage of wider range of machine learning models that can potentially perform better than the Random Forest.

BIBLIOGRAPHY

1. *Viswanathan S., Albert S. Kyle*. How to Define Illegal Price Manipulation. In: (2008).
2. Python webpage. url: https://www.python.org.

3. Scikit-Learn webpage. url: https://scikit-learn.org/stable/
4. Statsmodels webpage. url: https://www.statsmodels.org/stable/index.html.
5. *Jiahua Xu.* The Anatomy of a Cryptocurrency Pump-and-Dump Scheme. In: (2019).
6. *Haibo He Yunqian Ma.* Imbalanced Learning: Foundations, Algorithms, and Applications. In: (2013).
7. *Allen, F., and D. Gale.* (1992). Stock-Price Manipulation. The Review of Financial Studies 5 (3): 503–529.
8. *Aggarwal, Rajesh, and Guojun Wu* (2006). Stock Market Manipulations, Journal of Business 79 (4): 1915–1953.
9. *Williams-Grut, Oscar.* Market Manipulation 101: Wolf of Wall Street - Style Pump and Dump Scams Plague Cryptocurrency Markets, Business Insider. In: (2017).
10. *Kamps, Josh, and Bennett Kleinberg*, (2018), To the Moon: Defining and Detecting Cryptocurrency Pump-and-Dumps, Crime Science 7: 1–18.
11. *Foley Sean, Jonathan Karlsen, and Talis Putnins*, (2019). Sex, Drugs, and Bitcoin: How Much Illegal Activity Is Financed through Cryptocurrencies? Review of Financial Studies 32 (5): 1798–1853.
12. *Tao Li, Donghwa Shin, and Baolian Wang*. Cryptocurrency Pump-and-Dump Schemes. In: (2019).
13. *Trevor H., Robert T. and Jerome F*. The Elements of Statistical Learning. In (2001).
14. Aurélien Géron, Hands-on Machine Learning with Scikit-Learn, Keras and Tensorflow. In: (2019).
15. *Charu C.Aggarwal*, Data Classification, Algorithms and Applications. In: (2015).
16. *King. C.Fu*, Sequential Methods in Pattern Recognition and Machine Learning. In (1968).
17. *Scott Hartshorn*. Machine Learning With Random Forests And Decision Trees. In: (2016).

ОБНАРУЖЕНИЕ И ПРЕДСКАЗАНИЕ СЛУЧАЕВ НАКАЧКИ И СБРОСА ПОСРЕДСТВОМ ПРИМЕНЕНИЯ МОДЕЛЕЙ МАШИННОГО ОБУЧЕНИЯ

О. Зограбян, Г. Григорян

Ереванский Государственный Университет
Шанхайский Университет Экономики и Финансож

АННОТАЦИЯ

В данной статье речь идет о популярности криптовалют, которые привлекают на рынок все больше людей. Такая популярность и рост числа новичков на рынке создают благодатную почву для мошенников, чтобы обмануть общественность. Один из популярных способов мошенничества – это новое формирование консервативных схем pump-and-dump. В настоящее время схемы накачки и сброса привлекли внимание регуляторов криптовалюты и бирж, пытающихся заранее предсказать события pump-and-dump, чтобы сделать обмены более безопасными для общей базы пользователей. Такой алгоритм может помочь им предотвратить события накачки и сброса, ограничивая объемы покупки и продажи монеты, которая, по прогнозам, будет накачана. В результате торговля криптовалютой потенциально может стать более безопасной.

Ключевые слова: машинное обучение, проблемы классификации, финансовый анализ, крипторынок, события накачки и сброса.

УДК004.05

Поступила: 04.02.2021г.
Сдана на рецензию: 11.02.2021г.
Подписана к печати: 24.02.2021г.

ENABLING IN-FIELD OPERATION PROGRAMMABILITY FOR BUILT-IN SELF-TEST

A. Babayan

National Polytechnic University of Armenia

ababaya@synopsys.com

ABSTRACT

This paper describes Built-in self-test (BIST) solution with in-field operation programmability. This feature allows to have flexible test operation set for BIST. This can be configured after design stage of System on Chip (SoC). Thus, the custom algorithms, which could not be used for solutions with fixed operation set, are now available through this method.

Keywords: Memory Testing, System-on-Chip, Programmable Operations.

1. Introduction

The reliability of the memories in integrated circuits (IC) is one of the most important factors for the normal behavior of systems on chip (SoC). Different solutions, in general a BIST, were made to increase that reliability [1-3]. Usually, BIST solution is a processor, which has control over the memories in an IC, has control logic for the test sequence execution, coordinates its possible configurations and programming with other intellectual properties (IP) through provided interface.

To test a certain type of memory, the possible defects of that should be investigated and corresponding fault models should be defined [4]. After

the definition of that fault models, the algorithm, which can detect required types of such faults, is defined. The algorithm consists of operations, which drive the memory ports. The operation set is hardwired in current implementations of BIST processors, it is defined during design stage and some combinations cannot be applied on the memory ports. In other words, there is no possibility to declare new test operations after production. The more operations in BIST processor, the wider possibilities for testing memories. The programmability of these operations allows to have flexibility during execution of an algorithm, which can be useful not only for testing memories but also for their diagnosis.

2. BIST components

2.1. Programmable array for operations

At first, it is necessary to understand what the programming of an operation means.

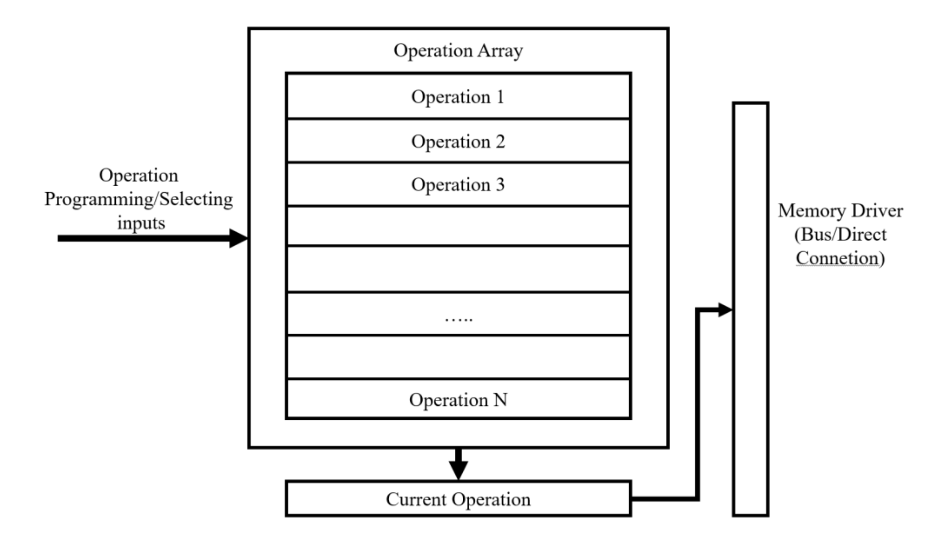


Fig 2. Operation Array Block Diagram.

Operation array consists of words (registers), which carry the code of operations, the input multiplexer which enables the connection to a word, and the register, which carries the reference (address) value of the corresponding word. The presence of such a structure increases the area of the processor depends on array size.

In Fig. 2, operations 1, 2, 3,…, N (N can be defined by the user at design stage) are responsible for enabling or setting some bits in final operation register, which drives the ports of the testing memory. The single operation element, carrying certain code, configures the memory driver in such a way, how it is loaded in the final operation register with specified format. The size of the element in array is determined by considering the memory ports, which should be driven, data patterns and other factors as well. It is not required to have all these descriptors in single operation cell, which means, that, for example, data pattern can be common for all these cells. As an example, for simple write operation with solid pattern (all "0"s/"1"s) the operation register should be configured in this way (the enabling of one signal is done by its setting logical "1")

OP_WRITE_0 = 000…000_0_1

Usually, operations have their own names, which is more understandable to users and it is more convenient to work with them.

In this example, the least significant bit (LSB) is directly connected to write enable bus of the processor. The nextbit is responsible for data pattern (for solid pattern one bit can be expanded by length of the bus data size, for other patterns it becomes larger). The data pattern can be generated by another component, so the existence of the corresponding bit(s) is(are) optional. Below scheme shows a brief view of one operation structure (Fig. 3).

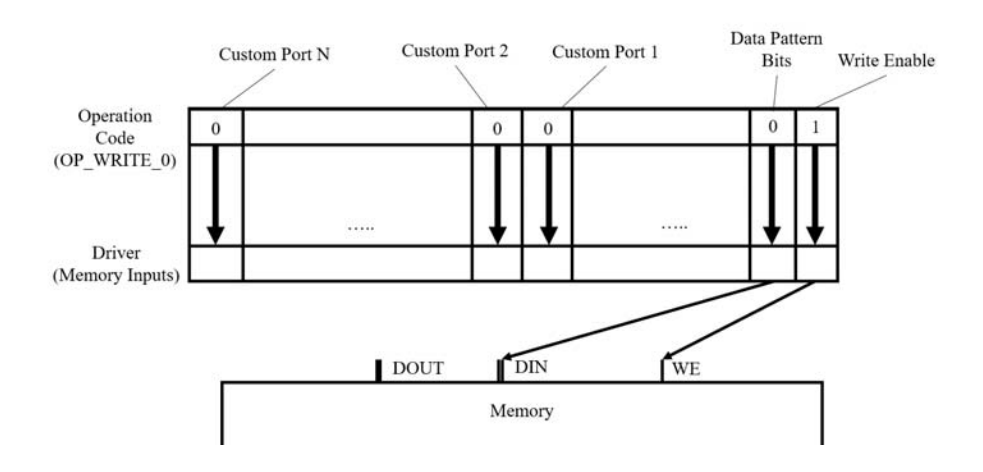


Fig 3. Write Operation Configuration.

The operation selection and programming are done through multiplexers, so, to make a march algorithm, only operation references (positions) are used.

Besides the ports, which are driven by processor, some flags (for example comparison flag) should be set when reading data from the memory. If operation contains comparison event, then the corresponding flag is transmitted to the comparator which gives an information about comparison. For operations, which contain comparison event, the expected data is givenconsidering the data pattern and special memory ports which can affect during write operation.

An 8x32 memory (Fig. 4) will be used as an example. Its portsare described Table 1.

Table 1.

Pinout description for memory 8x32.

Port Name	Description
WE	Write Enable
ME	Memory Enable

D	Input Data
ADDR	Address
BE	Byte Enable
Q	Output Data
P1…PN	User Specified Ports

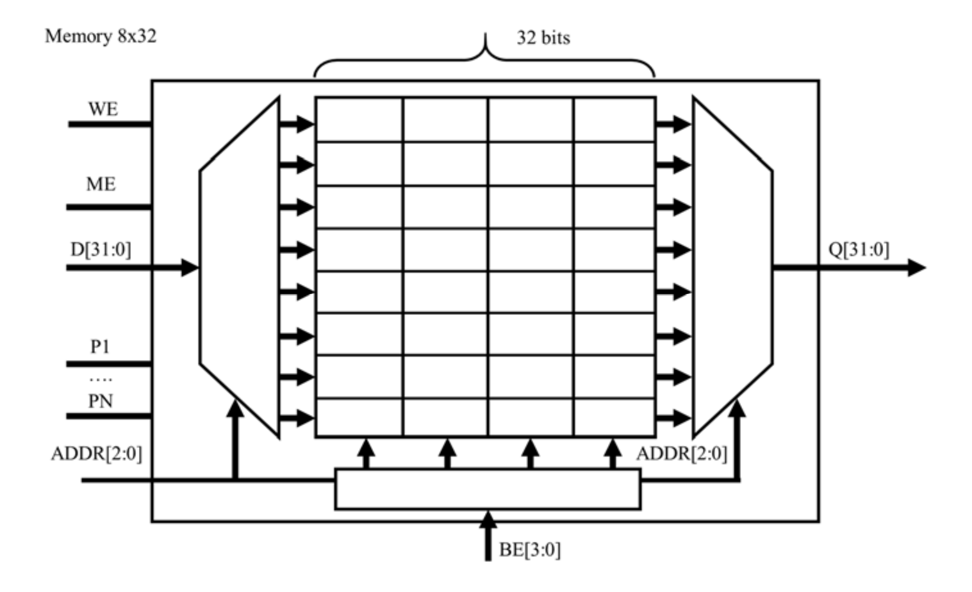


Fig 4. Memory 8x32.

Using the different approaches to implement the test environment for the memory [5–6], the final view of the system is presented in Fig. 5. The description of the operations for this memory is presented in Table 2.

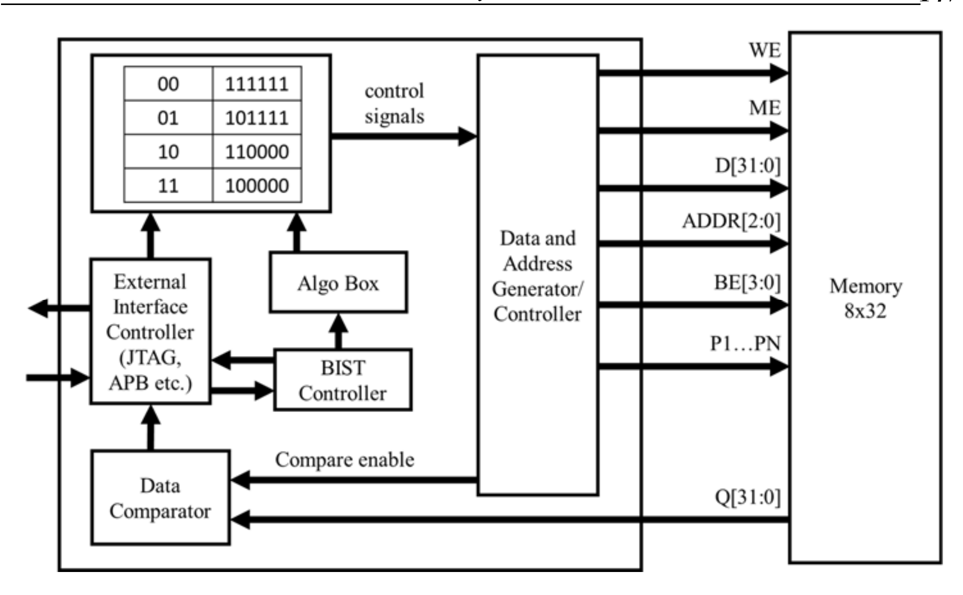


Fig 5. The usage of the programmable array for the operations for the memory 8x32.

Table 2.

Pinout description for the memory 8x32

Operation Name	ME	WE	BE	Operation address	Operation pattern
OP_WRITE_<D/!D>	active	active	all inactive	00	1_1_1111
OP_READ <D/!D>	active	inactive	all inactive	01	1_0_1111
OP_WRITE_B<D/!D>	active	active	all active	10	1_1_0000
OP_READE_B<D/!D>	active	inactive	all active	11	1_0_0000

3. Conclusion

The wide range of the operations allows to have fully control over testing memories which can be reached by the suggested programmable

operation array. This feature allows to have operations after the design stage and change these operations for the different test sequences by increasing the areaof the processor.The operation format is described. The working principle of this array and the execution of the operations are introduced.

REFERANCES

1. *Singh A., Kumar G.M., Aasti A.* Controller Architecture for Memory BIST Algorithms”, 2020 IEEE International Students' Conference on Electrical,Electronics and Computer Science (SCEECS), Bhopal, India, 2020. PP. 1–5.
2. *Manasa R., Verma R.and Koppad D.* Implementation of BIST Technology using March-LR Algorithm, 2019 4th International Conference on Recent Trends on Electronics, Information, Communication & Technology (RTEICT), Bangalore, India, 2019 .PP. 1208–1212.
3. *Harutyunyan G., Shoukourian S., Zorian Y.* Fault Awareness for Memory BIST Architecture Shaped by Multidimensional Prediction Mechanism, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Volume 38, Number 3, March 2019. PP. 562–575.
4. *Goor van de A. J.* Testing semiconductor memories, theory and practice, John Wiley & Sons, Inc., USA, 1991.
5. *Balch M.* Complete Digital Design: A Comprehensive Guide to Digital Electronics and Computer System Architecture, The McGraw-Hill Companies, Inc., USA, 2003.
6. *Hakhumyan A., Harutyunyan G., Shoukourian S., Vardanian V., Zorian Y.* Testing Electronic Memories based on fault and test Algorithm Periodicity, Patent No: US 9,831,000, Date of Patent: November 28, 2017, Appl. No. 14/484, 736, filed on September 12, 2014.

СОЗДАНИЕ ВОЗМОЖНОСТИ ПРОГРАМИРОВАНИЯ ОПЕРАЦИИ ДЛЯ ВСТРОЕННОГО САМОТЕСТИРОВАНИЯ

А.В. Бабаян

Национальный Политехнических Университет Армении

АННОТАЦИЯ

В данной статье представляется встроенное самотестирование с возможностью программирования операции. Эта функция позволяет иметь гибкий набор операций тестов для самотестирования. Данный набор можно настроить после этапа проектирования системы на кристалле (SoC). Таким образом, пользовательские алгоритмы, которые нельзя было использовать для решений с фиксированным набором операций, теперь доступны через данный метод.

Ключевые слова: самотестирование встроенной памяти, система на кристалле, программируемые операции.

УДК 004.005

Поступила: 19.04.2021г.
Сдана на рецензию: 26.04.2021г.
Подписана к печати: 29.04.2021г.

БЕНЧМАРКИНГ ИНСТРУМЕНТОВ СРАВНЕНИЯ БИНАРНОГО КОДА ПРОГРАММ

М.С. Арутюнян[1], Р.А. Оганнисян[2], Х.С. Смбатян[2]

Российско-Армянский Университет[1],
Ереванский Государственный Университет[2]

arutunian@ispras.ru, hr.hovhanisyan@ispras.ru, smbatyan1729@gmail.com

АННОТАЦИЯ

В данной статье рассматриваются инструменты сравнения бинарного кода программ, которые широко используются для анализа уязвимостей, поиска вредоносного кода, обнаружения нарушений авторских прав и т.д. В статье рассматриваются лучшие по результатам три инструмента известных на момент – BCC, Bin Diff, Diaphora. Инструменты основаны на статическом анализе программ. В качестве входных данных получают две версии программы в бинарной форме и сопоставляют их функции. Целью статьи является оценка качества вышеупомянутых инструментов. Для этого нами разработана система тестирования для автоматического определения точности и полноты каждого инструмента. F1 – мера на разработанной системе тестирования для инструмента BCC – 85.6%, для BinDiff – 82.4%, для Diaphora – 64.7%.

Ключевые слова: анализ бинарного кода, BCC, BinDiff, Diaphora.

Введение

Инструменты сравнения бинарного кода программ служат для сравнения двух версий программ с целью выявления их сходства и различий. Сравнение может быть на уровне базовых блоков, функций или целых программ. Часто исходный код программы недоступен, поэтому

в таких случаях сравнение двоичного кода имеет основополагающее значение.

Выявление схожести и различий исполняемых файлов является сложной задачей, поскольку из-за процесса компиляции может теряется (в зависимости от оптимизаций) большая часть семантики программы, включая имена переменных, имена функций и определения структур данных. Более того, результирующий двоичный код может значительно измениться при использовании разных компиляторов, разных оптимизаций и выборе различных целевых операционных систем и архитектур.



Рис 1. Определение пар функций, которые в идеальном случае должны найти инструменты.

Сравнение бинарного кода программ имеет широкий спектр применения – такой, как поиск ошибок, обнаружение вредоносных программ, их происхождение и кластеризация, автоматическая генерация исправлений, анализ исправлений, перенос информации между версиями программ и обнаружение нарушений авторских прав программного обеспечения.

Существуют много работ, посвященных анализу изменений программ. В статье представлен обзор и сравнение инструментов BinDiff (T. Dullen & R. Rolles, 2005).

Diaphora (Koret) и BCC (Aslanyan, Avetisyan, Arutunian, Keropyan, Kurmangaleev, & Vardanyan, 2017), которые на данный момент показывают наилучшие результаты. Для сравнения инструментов реализована тестовая система, с помощью которой можно выявить, насколько хорошо работают инструменты.

Обзор инструментов

В инструменте BinDiff (версия 6) [1] реализованы разные метрики для сопоставления функций. Метрики считаются на основе графов потока управления и графов вызовов функций. В работе авторы предлагают метод вычисления (MD-index) хеша для графов. Далее MD-Index используется для вычисления метрик на графах.

Инструмент Diaphora (версия 2.0.3) [2] сопоставляет функции с помощью различных наборов эвристик («лучшие сопоставления», «частичные и ненадежные сопоставления», «ненадежные сопоставления»). Эвристики применяются по очереди; если после применения текущей эвристики остаются несопоставленные функции, то применяется другая функция.

Работа инструмента BCC [3] делится на два этапа. На первом этапе происходит генерация графов зависимостей программы (ГЗП) и графа вызовов функций. На втором этапе происходит сопоставление функций на основе сгенерированных графов. Алгоритм сопоставления функций состоит из двух этапов. Сначала функции сопоставляются при помощи ряда эвристик. Например, пары функций сопоставляются, если хеши ассемблерного кода одинаковы. Те функции, которые не сопоставлялись с помощью эвристик, сопоставляются с помощью алгоритма определения максимального общего подграфа ГЗП.

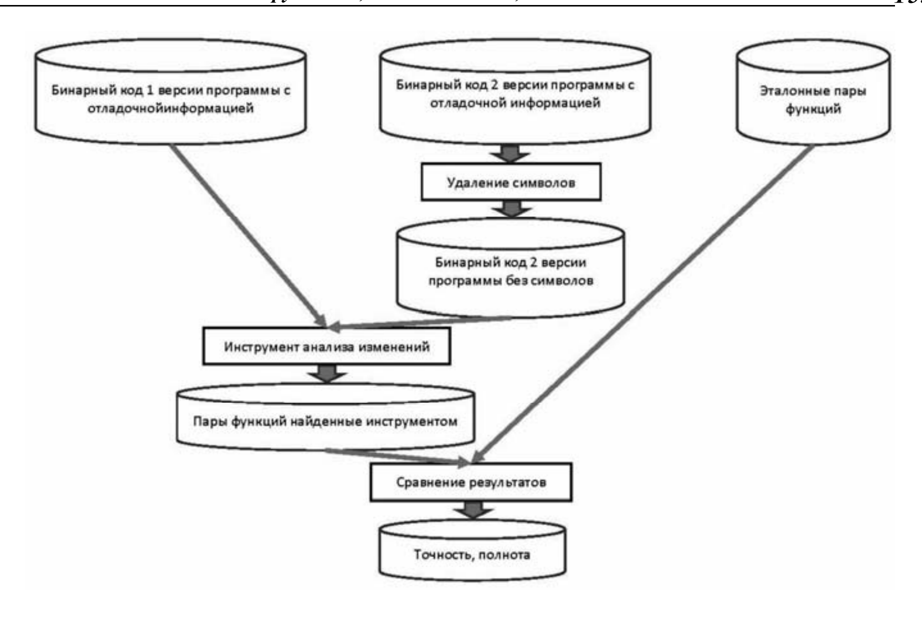


Рис. 1. Определение точности и полноты инструментов.

Схема тестовой системы

Схема тестовой системы состоит из двух частей – автоматическая генерация тестов и тестирование инструментов с использованием генерированных тестов. На Рис. 1 представлена схема автоматической генерации тестов. Для этого используются две версии (не обязательно разные) одной и той же программы. Сначала исходный код обоих версий компилируется с отладочной информацией, что позволяет сохранить имена символов, в частности, функций в базе данных. Компиляция может выполняться разными флагами оптимизаций. Далее из бинарного кода программ сопоставляются функции, которые имеют одинаковые имена. Полученный результат содержит пары функций, которые должны найти инструменты анализа изменений в идеальном случае.

Во второй части тестовой системы (Рис. 2) запускаются инструменты на тестах и считывается точность и полнота. Точность и полнота определяются следующим образом:

$$\text{Точность} = \frac{\text{количество правильно найденных результатов инструмента}}{\text{количество всех найденных результатов инструмента}}$$

$$\text{Полнота} = \frac{\text{количество правильно найденных результатов инструмента}}{\text{количество всех правильных результатов}}$$

Первым шагом является сохранение отображения (в базе данных) начальных адресов и имен функций для бинарного кода второй версии программы. Далее, из него удаляются символы (также удаляются имена функций) с помощью утилиты stripиз coreutils [4], полученный бинарный файл и бинарный файл первой версии программы передаются инструменту анализа бинарного кода. Целью удаления символов из бинарного файла является максимальное приближение генерированных тестов к реальным примерам, так как такая информация почти всегда недоступна в анализируемых программах.

Инструменты сравнения бинарного кода программ выдают сопоставленные пары функций, которые сравниваются с парами, полученными из первой части тестовой системы. В системе тестирования автоматически разбираются результаты инструментов и приводятся к единой форме. Для каждой найденной пары функций (f, f') восстанавливается оригинальное имя f', используя ее адрес, и база данных, в которой заранее хранили отображение начальных адресов и имен функций для бинарного кода второй версии программы. Если оригинальное имя f' является f, то результат считается истинно-положительным, в противном случае – ложноположительным. На основе полученных результатов считаются точность и полнота.

Результаты

Результаты получены на 105 программах, входящих в набор coreutils [4]. Исходные файлы были скомпилированы компиляторами g++ [5] и clang++ [6] с разными флагами оптимизаций для x86–64

архитектуры. В Табл. 1 приведены результаты инструмента BCC, BinDiff и Diaphora:

Версия coreutils 1	**Версия coreutils 2**	**Компилятор, оптимизации 1**	**Компилятор, оптимизации 2, strip**	**BCC средняя точность**	**BCC средняя полнота**	**BinDiff средняя точность**	**BinDiff средняя полнота**	**Diaphora средняя точность**	**Diaphora средняя полнота**
8.29	8.30	g++ o0	g++ o0	97.8%	98.2%	97.8%	97.8%	97.6%	72.5%
8.29	8.30	g++ o2	g++ o2	97%	82.2%	96.6%	81.4%	96.3%	52.8%
7.6	8.30	g++ o0	g++ o0	72.1%	69.5%	62.9%	62.9%	92.5%	35.5%
7.6	8.30	g++ o2	g++ o2	85.4%	77.9%	83.4%	76.5%	95.1%	46.5%
8.30	8.30	clang++ o0	clang++ o0	98.8%	98.3%	98.8%	97.9%	98.3%	75%
8.30	8.30	clang++ o2	clang++ o2	97.2%	85.6%	98.3%	85.9%	97.5%	59.2%
8.30	8.30	clang++ o0	clang++ o2	79.3%	67.4%	72.2%	61.5%	93.1%	34.3%
8.30	8.30	clang++ o2	clang++ o3	90.7%	81.2%	86.6%	76.6%	97.2%	41.2%
8.30	8.30	g++ o0	g++ o0	98.8%	98.3%	98.8%	97.9%	98.3%	74.7%
8.30	8.30	g++ o2	g++ o2	98.2%	82.4%	98.2%	81.7%	97.3%	55.2%
8.30	8.30	g++ o0	g++ o2	84.8%	69.2%	75.5%	63.7%	91.3%	34.5%
8.30	8.30	g++ o0	clang++ o0	94.9%	89.4%	91.2%	85.4%	95.6%	38.1%
8.30	8.30	g++ o2	clang++ o2	89.1%	76.1%	81.1%	71%	93.8%	36.7%
8.30	8.30	g++ o0	clang++ o2	78.9%	67.7%	71.2%	61.5%	93%	34.5%
8.30	8.30	g++ o2	g++ o3	90%	77%	88.7%	75.4%	93.5%	44.8%
В среднем				**90.2%**	**81.4%**	**86.8%**	**78.5%**	**95.4%**	**49.0%**

Из таблицы можно увидеть, что полнота результатов инструмента BCC превышает конкурентов. Наилучшую точность показывает инструмент Diaphora, однако полнота результатов инструмента намного меньше остальных. В среднем F1-мера (2*точность*полнота / (точность + полнота)) для BCC – 85.6, для BinDiff – 82.4, для Diaphora – 64.7. Притом разница F1-мер инструментов больше, когда разница между версиями анализируемых программ бо́льшая (много изменений) или они скомпилированы разными опциями компиляции.

Заключение

В статье сравниваются три инструмента анализа изменений программ. Выбор этих инструментов обусловлен тем, что они поддерживаются, широко используются и показывают наилучшие результаты. Сравнение проводилось на ряде программ набора coreutils. Результаты показали, что точность и полнота инструментов высока, когда разница версией между анализируемыми программами меньше и когда они компилированы одними и теми же оптимизациями компилятора.

ЛИТЕРАТУРА

1. *Aslanyan H., Avetisyan A., Arutunian M., Keropyan G., Kurmangaleev S., & Vardanyan V.* (2017). Scalable Framework for Accurate Binary Code Comparison. *2017 Ivannikov ISPRAS Open Conference (ISPRAS)*. СС. 34–38). Moscow.
2. Clang: a C language family frontend for LLVM. (б.д.): https://clang.llvm.org/
3. Coreutils - GNU core utilities. (б.д.): https://www.gnu.org/software/coreutils/
4. GCC, the GNU Compiler Collection. (б.д.): https://gcc.gnu.org/
5. *Koret, J.* (б.д.). Получено из https://github.com/joxeankoret/diaphora
6. *Dullien T. & Rolles R.* (2005). Graph-based comparison of executable objects. Symposium sur la Securite des Technologies de l'Information et des Communications.

BENCHMARKING OF PROGRAMS BINARY CODE COMPARISON TOOLS

M. Arutunian, H. Hovhannisyan, Kh. Smbatyan

Russian-Armenian University
Yerevan State University

ABSTRACT

Binary code comparison tools are widely used to analyze vulnerabilities, search for malicious code, detect copyright violations, etc. The article discusses the best three tools known at the time – BCC, BinDiff, Diaphora. The tools are based on static analysis of programs. Tools receive as input data two versions of the program in binary form and match their functions. The purpose of the article is to assess the quality of the tools. For this, we developed a testing system to automatically determine precision and recall of each instrument. F1-score on the developed testing system for BCC instrument is 85.6%, for BinDiff – 82.4%, for Diaphora – 64.7%.

Keywords: binary code analysis, BCC, BinDiff, Diaphora.

ФИЗИКО-ТЕХНИЧЕСКИЕ НАУКИ

УДК 621.396.673

Поступила: 30.03.2021г.
Сдана на рецензию: 08.04.2021г.
Подписана к печати: 22.04.2021г.

METHOD OF INCREASING ELIPTICAL POLARIZED ANTENNA AXIAL RATIO

A. Stepanyan, T. Grigoryan, D. Ghazaryan, L. Khachatryan

National Polytechnic University of Armenia,
Institute of Information and Telecommunication Technologies and Electronics

araratstepanyan9@gmail.com, grigoryan970@gmail.com,
david.ghazaryan.no1@gmail.com, khachatryan-l@inbox.ru

ABSTRACT

In a number of radio engineering systems, the use of circularly polarized antennas has become extremely important and relevant. Currently various solutions are being proposed to create antennas with a new type of circular polarization and to increase the axial ratio of existing antennas. It is also necessary to create such antennas that will have a small size and mass and will be integrated into radio engineering circuits. A microstrip antenna with elliptical polarization was investigated and designed, from which circular polarized antenna with dual feed has been obtained. The design was carried out in the FEKO software environment. The calculated results are presented in the form of graphs. The developed antenna can be used in 5G system.

Keywords: circularpolarization, microstripantenna, dual feed, axialratio, FEKO environvent, 5G.

Introduction

Microstrip antennas are flat resonators, that include dielectric, earthing plane, radiant surface and feed chain with permittivity ε_r and thickness$h$.These are the advantages of such antennas: they are small, convenient, and do not take up expensive resources. Disadvantages: low efficiency, high quality, inaccurate polarity behavior, narrow band, due to high quality, size in the low frequency range is very large and it is advisable to use the antenna in the microwave range. Circular polarized antennas are widely used in satellite communication and global positioning system (GPS). The rationale for use in such systems is that the characteristics of circular polarity eliminate the effects of multi-channel spread and provide a fixed position angle between the transmitter and the receiver. The advantages of circular polarity: the receiver antenna always accepts a signal component regardless of the angle of its position, it is more stable to atmospheric extinctions, there is no need for direct visibility and it easily bypasses the barriers [1].

Type of this antenna and other circularly polarized antennas tend to suffer from high cross-polarization levels.

Nowadays it's process different methods for increasing microstrip antenna axial ratio.

Methodology

It's researched and designed notched circular patch antenna (Figure 1). Effective radius of the patch:

$$a_{eff} = \frac{X'_{np}c}{2f\sqrt{\epsilon_r}} \quad (1),$$

where X'_{np} are the zeros of the derivative of the Bessel function $J_n(x)$ of order n, as is true of TE-mode circular waveguides. The lowest-order mode, TM_{11} uses

$$X'_{11} = 1.84118$$

.

Antenna physical radius:

$$a = \frac{a_{eff}}{\sqrt{1+\frac{2h}{\pi a \varepsilon_r}(\ln\left(\frac{\pi a}{2h}\right)+1.7726)}} \quad (2),$$

where h is the substrate thickness [2].

Microstrip antenna was designed for 3 GHz frequency. As a dielectric, the FR4 brand was chosen, the parameters of which are $\varepsilon_r = 4{,}6$; $h = 1{,}6$mm; $tg\delta = 0{,}017$.Calculating the antenna radius by 1: $r = 13{,}12$mm.

A dual feed circuit has been built, which is a quadrature hybrid directional coupler. Such a circuit provides matching between the antenna and the generator for a given frequency and the phase difference of the signal at the outputs is 90 degrees. Dimensions of the quadrature coupler: $w_0 = 2{,}6$mm (line with a impedance of 50 Ω); $w_1 = 3{,}5$ mm (line with a impedance of 35,35 Ω) [3]. Thus, the antenna is fed in two ways (Figure 1), between which the phase difference is 90 degree. So, the radiated electric field vectors create a circular polarity.

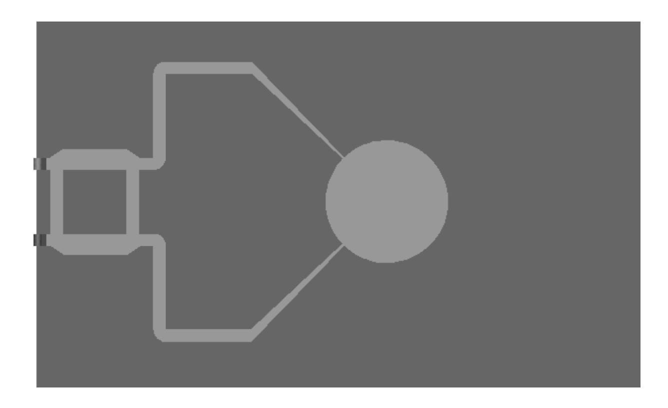


Figure 1. Structure of dual feed microstrip antenna.

By dual feeding, the input impedance of the antenna changes and losses increase, because of which the antenna gain factor decreases. It is necessary to coordinate the resistance. The matching was carried out by a linear matching circuit. To calculate the size of the transformer in the software environment, two directions of the antenna input impedance were solved. Thus, the impedance of 50 Ω must be matched with the resistance of 100 Ω. Transformer dimensions: $l = 16$ mm, w_{top} =2,7 mm, w_{bottom} =0,5 mm. The total size of the antenna: 11,5x7 cm.

Results

The main characteristics of the developed antenna obtained in the FEKO software environment are presented [4]. Figure 2 shows the frequency dependence of the microstrip antenna's voltage standing wave ratio (VSWR) with sided feeding. In the 2,98…3,03 GHz frequency range: VSWR$<$ 2.As shown figure 3 antenna axial ratio maximum value is $-5,3$ dB at 3 GHz.

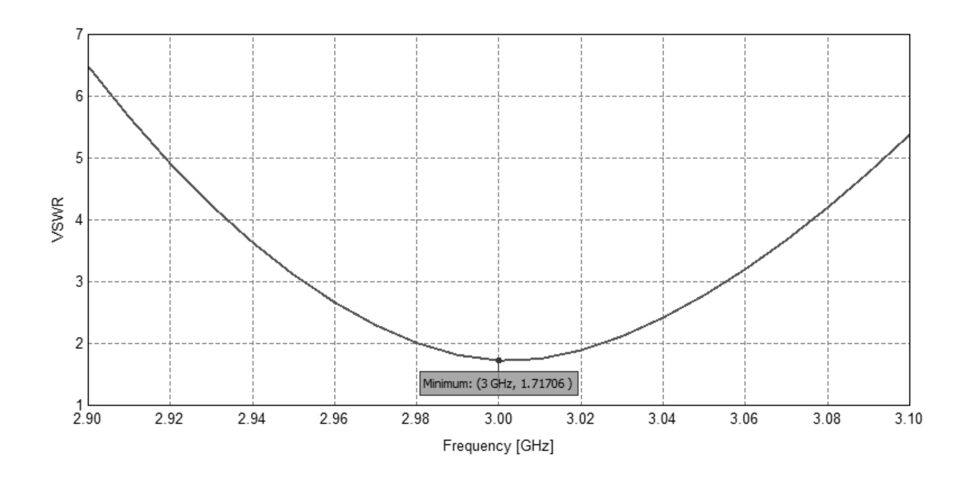


Figure 2. Frequency dependence of the VSWR of the microstrip antenna.

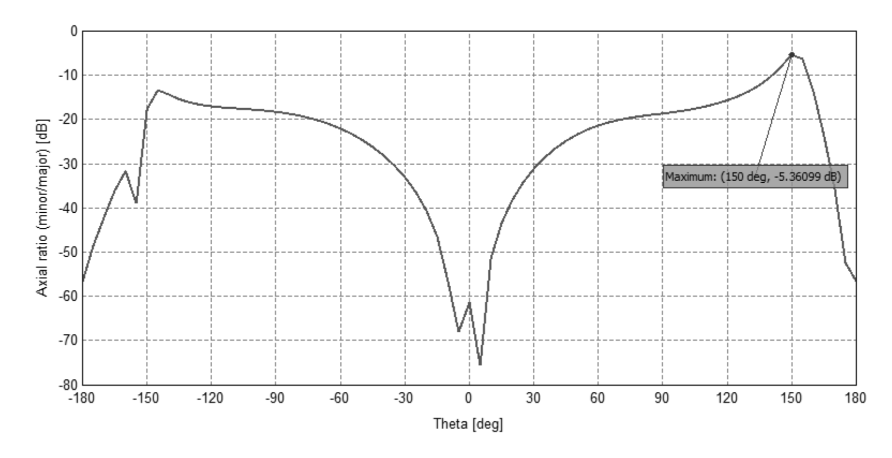


Figure 3. Antenna axial ratio at 3 GHz frequency.

Figure 4 shows a graph of the frequency dependence of the VSWR of a dual feed antenna, and Figure 5 shows the frequency dependence of the antenna gain on the main radiation direction. $2{,}95 \ldots .3{,}362$ GHz in the frequent range VSWR<2, Gain>0 dB. The impedance bandwidth approximately doubles compared to the singly fed patch. At a frequency of 3 GHz: Gain $=$ 2.5 dB. The main lobe width in θ plane is $2\theta_{0.5} = 80{,}5°$, in φ plane is $2\varphi_{0.5} = 77{,}15°$ (Figure 6).

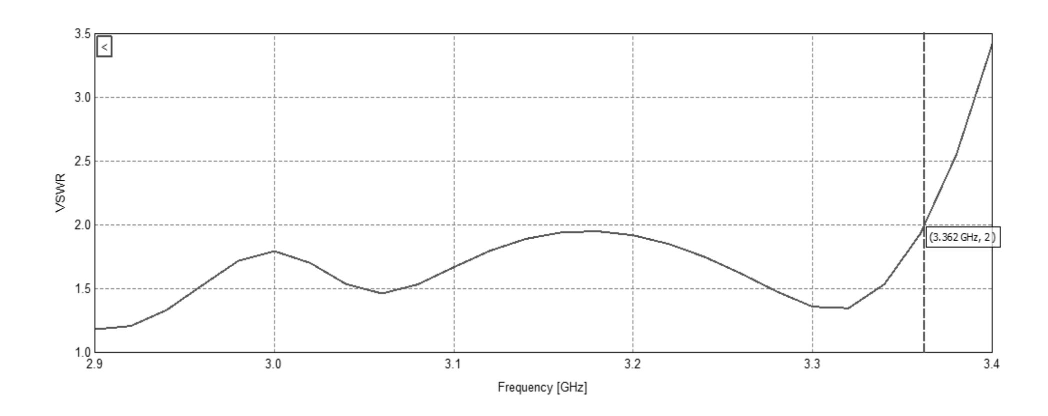


Figure 4. Dual feed antenna VSWR-frequency dependency.

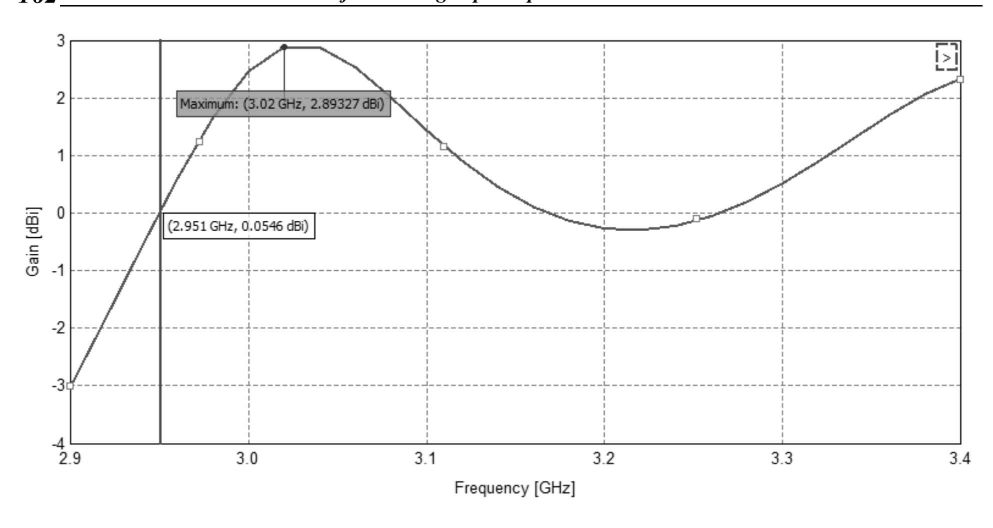


Figure 5. Frequency dependence of the antenna gain in the main radiation direction.

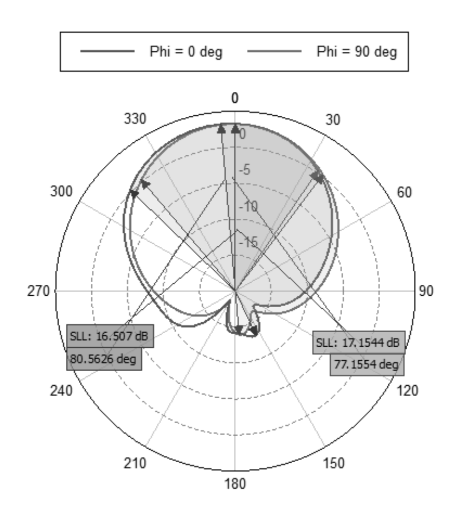


Figure 6. Radiation pattern of the antenna by dual feed at a frequency of 3GHz in the θ and φ planes.

Figure 8 shows the dependence of the axial ratio of dual feed antenna at 3; 3,16; 3,3 GHz frequencies from θ angle. The maximum value of the axial ratio of the antenna at a frequency of 3 GHz is -0,15 dB. The AR value

is close to zero, hence the antenna has a high circular polarized with an axial ratio.

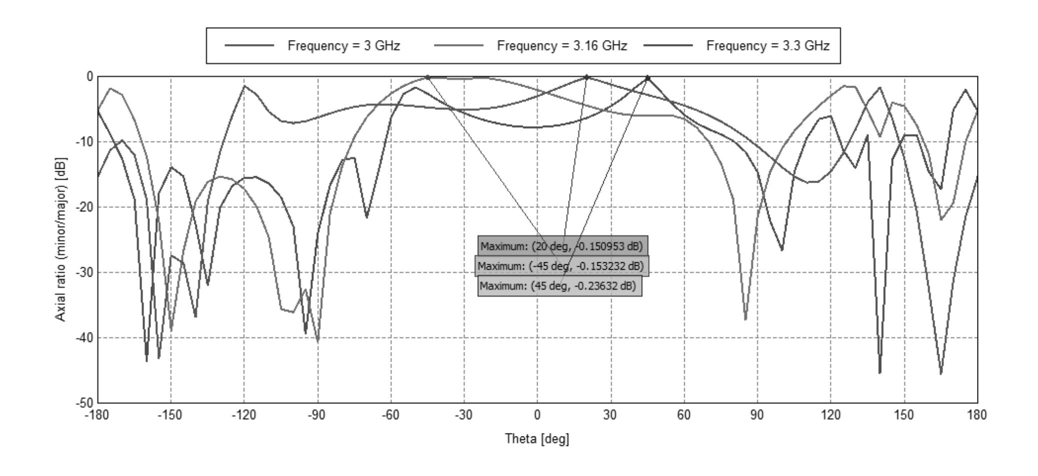


Figure 7. Antenna axial ratio θ dependency f or 3; 3,16 and 3,3 GHz.

Conclusion

- A microstrip antenna with elliptical polarity was studied and designed and from that a circular polarized antenna with dual feed was obtained.
- The operating frequency range of the antenna is 2.95...3.362 GHz, where the VSWR< 2.
- At a frequency of 3 GHz: Gain $= 2.5$ dB. The main lobe width in θ plane is $2\theta_{0.5} = 80{,}5°$, in φ plane is $2\varphi_{0.5} = 77{,}15°$.
- The maximum value of the axial ratio of the antenna at a frequency of 3GHz is -0,15dB. As a result, a circular polarized antenna with a high axial ratio is obtained.
- The antenna can be used in 5G mobile communication systems.

REFERENCES

1. *Thomas A. Milligan* Modern Antenna Design. - 2ndedition, John Wiley & Sons. 2005. 313p.
2. *Stepanyan A., Manukyan T., Eyramjan S.* Methods designing of circular polarization directional antenna// Proceedings of NAS RA and NPUA. 2019
3. *Pozar David M.* Microwave Engineering. 4-th edition, John Wiley & Sons. 2011. 147p.
4. *Atef Z. Elsherbeni, Payam Nayeri, Reddy C.* Antenna Analysis and Design Using FEKO Electromagnetic Simulation Software//SciTech Publishing, Edison, NJ. 2014.

МЕТОД ПОВЫШЕНИЯ КОЭФФИЦИЕНТА ЭЛЛИПТИЧНОСТИ АНТЕННЫ С ЭЛЛИПТИЧЕСКОЙ ПОЛЯРИЗАЦИЕЙ

А.Г. Степанян, Т.А. Григорян, Д.А. Казарян, Л.Х. Хачатрян

Национальный Политехнический Университет Армении
Институт Информатики и Телекоммуникационных технологий и Экономики

АННОТАЦИЯ

В данной статье речь идет о том, что в ряде радиотехнических систем использование антенн с круговой поляризацией стало чрезвычайно важным и актуальным. В настоящее время предлагаются различные решения для создания антенн с круговой поляризацией нового типа и увеличения коэффициента эллиптичности антенн. Также необходимо создать такие антенны, которые будут иметь небольшие размеры и массу и будут интегрированы в радиотехнические схемы. Исследована и сконструирована микрополосковая антенна с эллиптической поляризацией, из которой получена антенна с круговой поляризацией с двухпортовым питанием. Проектирование проводилось в программной среде FEKO. Результаты расчетов представлены в виде графиков. Разработанную антенну можно использовать в системе 5G.

Ключевые слова: круговая поляризация, микрополосковая антенна, с двухпортовым питанием, коэффициент эллиптичности, программная среда FEKO, 5G.

БИОЛОГИЯ

УДК 577.322; 004.852

Поступила: 23.02.2020г.
Сдана на рецензию: 03.03.2021г.
Подписана к печати: 24.03.2021г.

РАЗРАБОТКА ПРОГРАММЫ КОНФОРМАЦИОННОГО АНАЛИЗА И ВИЗУАЛИЗАЦИИ РЕЗУЛЬТАТОВ МОЛЕКУЛЯРНОГО ДОКИНГА С ИСПОЛЬЗОВАНИЕМ МЕТОДОВ КЛАСТЕРИЗАЦИИ

Ч.Г. Мовсисян, В.С. Камарян, Л.С. Унанян, А.С. Овакимян

Российско-Армянский Университет

movchinar@gmail.com, vghamaryan@rau.am, lernik.hunanyan@rau.am, anna.hovakimyan@rau.am

АННОТАЦИЯ

В данной работе представлены результаты создания удобного инструмента для обработки результатов молекулярного докинга с использованием методов кластеризации и визуализации. Одной из основных задач современной биоинформатики является анализ больших данных, полученных в результате медико-биологических исследований.

Ключевые слова: биоинформатика, молекулярный докинг, AutoDockvina, FOREL, кластеризация.

Введение

С каждым годом увеличивается количество биологических данных. Это является результатом быстро развивающихся экспериментальных и теоретических знаний о биообъектах, и в данном контексте адаптация и применение новых методов обработки больших данных считается одной из приоритетных задач современной биоинформатики [1]. При этом обработка полученных результатов является довольно трудоемким процессом, связанным с одновременным использованием различных программных пакетов и модулей, где выходные результаты отображаются в разных файловых форматах с использованием разных алгоритмов [2]. Автоматическая обработка, управление, анализ данных, полученных экспериментальным путем, существенно облегчают процесс интерпретации полученных результатов. Важнейшей задачей для решения данной проблемы является классификация и структуризация полученных результатов, ярким примером которого считается процесс кластеризации с использованием машинного обучения [3].

Кластерный анализ (кластеризация, самообучение, обучение без учителя или таксономия) используется для автоматического создания списка шаблонов с помощью обучаемых данных [4]. Все предметы, то есть подлежащие кластеризации данные представляются системе, не указывая, к какому типу они принадлежат. Предполагается, что данные, изучаемые в области свойств, состоят из ряда «шаров». Задача системы – выявить и сформулировать их. Это означает, что методы кластерного анализа позволяют разделить заданную группу данных на подгруппы подобных объектов, называемых «кластерами», «классами» или «таксонами» [5].

Одним из биоинформатических методов прогнозирования взаимодействия биологически активных соединений (лиганд) и биомакромолекулой (мишень) является метод молекулярного докинга [6]. Полученные пространственно-энергетические результаты молекулярного докинга являются оценочной функцией для понимания процессов

комплексообразования системы лиганд-мишень. В процессе обработки полученных результатов часто приходится сталкиваться с проблемой конвертирования файловых форматов, систематизацией полученной информации, визуализацией и т.д. [6].

Разработка универсального инструмента, приводящая к автоматизации процесса обработки и визуализации результатов *insilico* эксперимента, в частности, результатов молекулярного докинга, намного упраздняет процесс интерпретации данных, сохраняя при этом время и ресурсы. В данной работе представлены результаты создания удобного инструмента для обработки результатов молекулярного докинга с использованием методов кластеризации и визуализации.

Материалы и методы

Разработанная нами программа была реализована в среде Python [8]. Алгоритм кластерного анализа FOREL реализован без использования каких-либо библиотек. FOREL относится к классу таксономических алгоритмов [9]. Множество объектов, мощность которых равна m, можно разделить на k таксонов (k < m) несколькими различными способами. FOREL использует F-критерий, основанный на гипотезе компактности. Объекты, принадлежащие одному и тому же таксону, расположены близко друг к другу по сравнению с объектами, принадлежащими разным таксонам. В результате получаем шаровидные таксоны [10].

Определение геометрического расстояния объектов в многомерном пространстве проводился с использованием расчета Евклидового расстояния по уравнению 1.

$$d(x, x') = \left(\sum_{k=1}^{d} \left| x_k - x'_k \right|^q \right)^{\frac{1}{q}} \quad (1)$$

Для работы с файлами *.pdbqt использовался промежуточный модуль извлечения параметров наилучших конформеров с использованием библиотек BioPandas [11], Pandas [12], Numpy [13]. Визуализация кластеров проводилась с помощью библиотеки Matplotlib [14].

В качестве исходных данных были взяты результаты трех различных молекулярных докингов с образованием одного, двух и трех больших кластеров конформеров. Исследуемые молекулярные модели мишеней были взяты из банка данных RCSB [15] с соответствующими идентификационными номерами PDB ID: 1TBG, 6W9C, 4X7Q. В качестве лигандов были взяты молекулярные модели биоактивных соединений, предоставленные Научно-технологическим центром органической и фармацевтической химии НАН РА [16].

Для тестирования разработанной нами программы в качестве базовых параметров были взяты полученные результаты докинга для трех комплексов лиганд-мишень (Таблица 1).

Таблица 1.

Базовые параметры молекулярного докинга для трех взаимодействий

N	Мишень (PDB ID)	Количество сгенерированных конформеров	Количество конформеров удовлетворяющих условиям	Количество повторений эксперимента	Континуум исчислений	Средняя RMSD (Å)
1.	4X7Q	100	25	5	200	≤ 1,9
2.	6W9C	100	15	7	200	≤ 1,6
3.	1TBG	100	20	6	200	≤ 1,8

Молекулярный докинг проводился с использованием программы Auto Dock Vina [17], который имеет широкое применение в области биоинформатики и биомоделирования. В основе программы лежит Ламаркский генетический алгоритм. Одним из особенностей этого программного пакета является возможность проведения «слепого докинга», позволяющего изучать взаимодействие лиганд-мишень даже в том случае, когда активный центр мишени неизвестен. Прогнозирующий фактор данного программного пакета, иначе говоря, точность *insilico* эксперимента, достаточно высока [18].

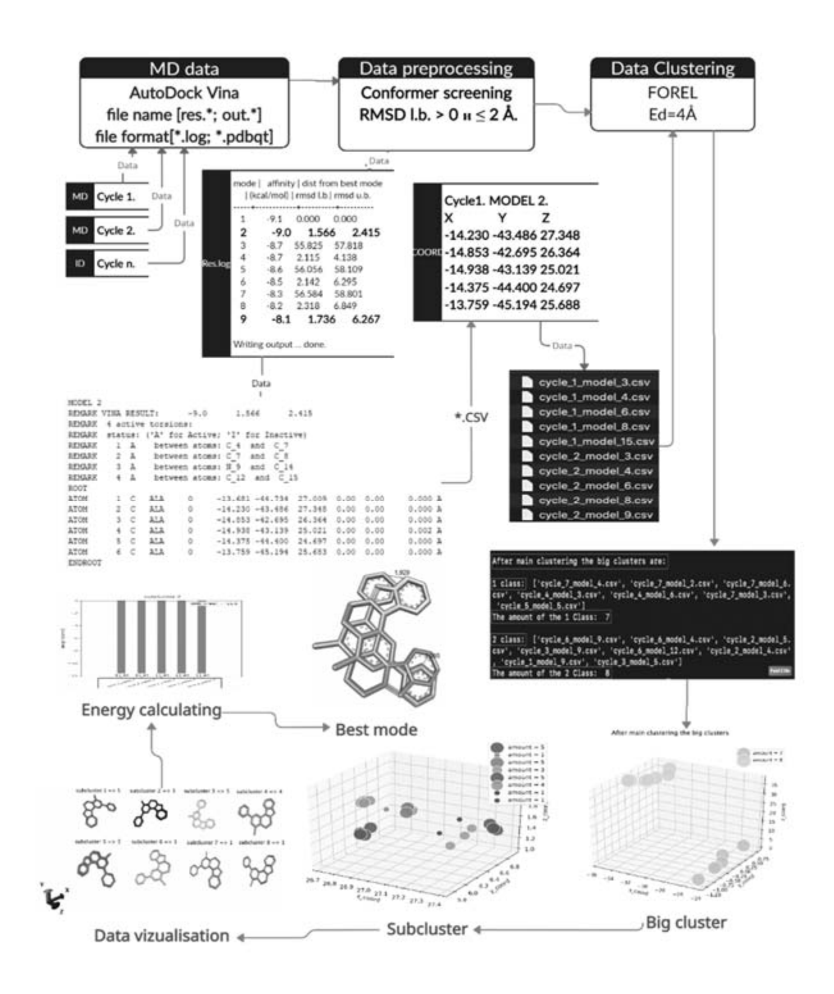


Рис. 1. Алгоритм разработанной нами программы.

Сравнительный анализ полученных пространственных параметров наилучших конформеров, выбранных на основе разработанной нами программы, был реализован с использованием программы BIOVIA Discovery Studio Visualizer [19]. В качестве оценочной функции идентичности конформеров был использован метод супер позиционирования [20] (контрольный метод). На Рис. 1 приведен алгоритм разработанной нами программы.

Результаты и обсуждения

Программа, разработанная нами, представляется набором из нескольких блоков. Первый блок – это исходные данные молекулярного докинга, которые отображаются двумя файлами с заданными парами форматов *.log и *.pdbqt. Результаты молекулярного докинга передаются в блок отбора (второй блок), где обрабатываются полученные данные. В результате этого этапа производится отбор лучших конформеров для последующего анализа. В качестве условия отбора используется значение среднеквадратического отклонения (RMSD l.b.>0 и ≤2 Å).

В результате для каждого цикла проводится отбор конформеров, соответствующих условию с дальнейшим извлечением из *.log. Следующим шагом проводится отбор пространственных параметров лучших конформеров из файлов *.pdbqt в каждом цикле с последующим перемещением в *.csv файл.

Таким образом, в результате работы блока отбора получаются ряд файлов *.csv, которые описывают положение атомов в трехмерном пространстве в результате взаимодействия лучших конформеров молекулярной модели лиганда по отношению к мишени. На этом этапе отбираются также энергетические значения взаимодействий лучших конформеров. Последний указан во 2-м столбце файла *.log. Задача основного блока (третий блок) состоит в том, чтобы выполнить кластерный анализ лучших конформеров, отобранных при помощи блока отбора.

На данном этапе каждый конформер считается отдельным объектом, в качестве характеристики описывающийся трехмерными координатами каждого атома. Используя алгоритм FOREL, определяется распределение лучших конформеров в трехмерном пространстве относительно друг друга. Сходство объектов определялось эвклидовым расстоянием, используя фиксированное расстояние в качестве критерия.

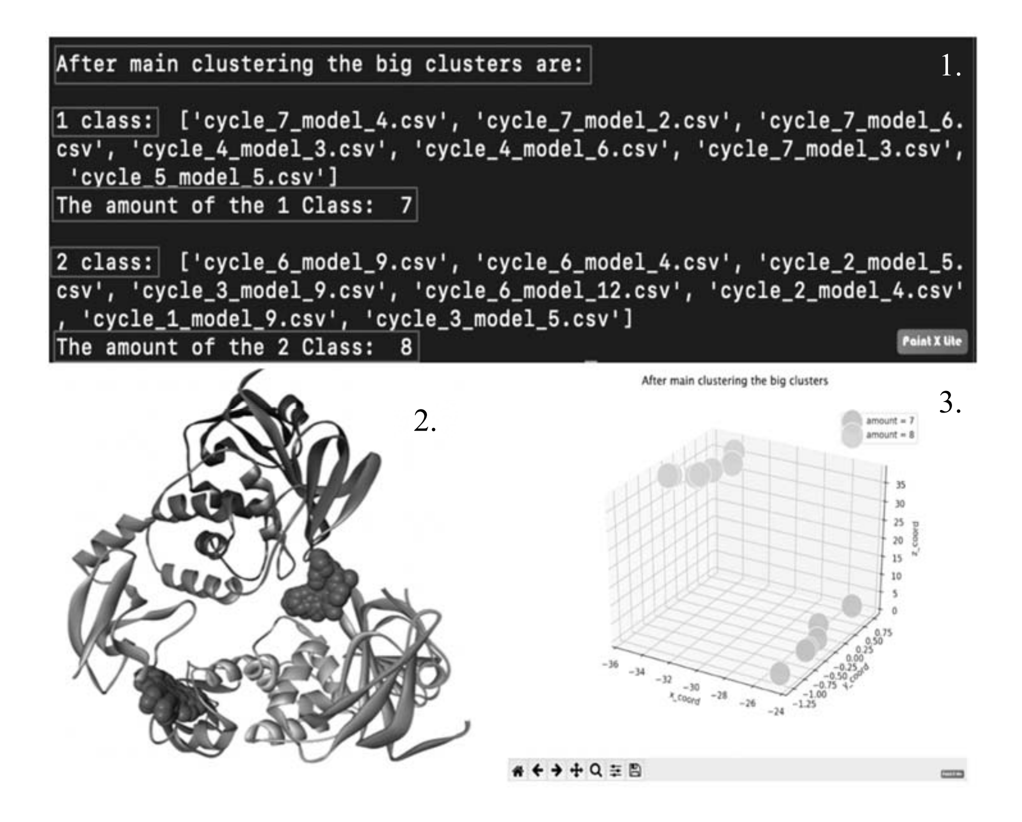


Рис. 2. Визуализация результатов кластеризации и получения больших кластеров. 1. Список лучших конформеров, входящих в большие кластеры. 2. Распределение конформеров в местах связывания (получена на основе BIOVIA Discovery Studio Visualizer). 3. Визуализация тех же конформеров, полученных при помощи нашей программы.

Необходимо отметить, что кластерный анализ с использованием алгоритма FOREL проводился в два этапа для достижения более точных результатов. На первом этапе для каждого из лучших конформеров вычислялся «центр тяжести» путем расчета среднеарифметического значения пространственных параметров о положении атомов (координаты X, Y, Z). Следующим шагом являются полученные значения «центров тяжести», где в качестве критерия фиксированного расстояния близости была выбрана дистанция в 4 Å. В результате были получены большие кластеры (bigclusters).

Вторым этапом рассматриваются сходства пространственных положений атомов конформеров, входящих в большие кластеры, где фиксированное расстояние близости равно 2 Å± 5%. В этом случае Эвклидово расстояние между атомами конформеров не превышают установленное расстояние. В результате в больших кластерах образуются схожие субкластеры (subclusters) (Рис. 3).

Получение информации о субкластерах является ключевым фактором для отбора наилучшего положения конформеров в большом кластере. Необходимо отметить, что визуализация результатов отбора конформеров осуществляется при работе основного блока с отображением в трехмерном пространстве. Известно, что пространственно-энергетические характеристики комплексообразования описывают структурно-функциональные особенности взаимодействия. Эти особенности позволяют идентифицировать наилучшие конформеры, имеющие определенное положение относительно мишени и высокие энергетические значения при комплексообразовании. В результате программа выбирает «конформеры-лидеры» на основе пространственно-энергетических характеристик комплексообразования и представляет пользователю для дальнейшей работы. Полученные нами результаты свидетельствуют, что кластеры и субкластеры, полученные путем кластеризации с использованием программы, практически идентичны с теми же результатами, полученных контрольным методом.

1.

```
After second clustering the subclusters are:
1 cluster's subclusters are:
[['cycle_4_model_6.csv', 'cycle_5_model_5.csv', 'cycle_7_model_6.csv'], ['
cycle_7_model_3.csv'], ['cycle_7_model_2.csv'], ['cycle_4_model_3.csv'], [
'cycle_7_model_4.csv']]
The amount of subclusters are:  5
2 cluster's subclusters are:
[['cycle_3_model_9.csv'], ['cycle_2_model_5.csv', 'cycle_6_model_12.csv',
'cycle_1_model_9.csv'], ['cycle_3_model_5.csv'], ['cycle_6_model_4.csv'],
['cycle_2_model_4.csv', 'cycle_6_model_9.csv']]
The amount of subclusters are:  5
```

2.

3.

4.

5.

Рис. 3. Результаты кластеризации для полученных субкластеров.
1. Список наилучших конформеров в субкласторе. 2. Отклонение пространственных параметров конформеров внутри субкластера, полученных контрольным методом. 3. Визуализация полученных субкластеров в трехмерном пространстве. 4. Вид субкластера на основе контрольного метода.
5. Качественно-количественные характеристики полученных субкластеров, входящих в большой кластер.

Таким образом можно констатировать, что разработанная нами программа полностью соответствует требованиям и поставленным задачам полной автоматизации многоступенчатого процесса обработки результатов молекулярного докинга путем кластеризации и визуализации. Данная программа может стать удобным инструментом для быстрого и точного анализа результатов, полученных молекулярным докингом, при этом сохранить затраты на ресурсы и время, что является немаловажным.

ЛИТЕРАТУРА

1. *Merelli, I., Pérez-Sánchez, H., Gesing, S., & D'Agostino, D.* (2014). Managing, analysing, and integrating big data in medical bioinformatics: open problems and future perspectives. BioMed research international, 2014.
2. *Ohlebusch, E.* (2013). Bioinformatics Algorithms. Enno Ohlebusch.
3. *Witten, I.H., Frank, E., Hall, M.A., & Pal, C.J.* (2005). Practical machine learning tools and techniques. Morgan Kaufmann, 578.
4. *Kassambara, A.* (2017). Practical guide to cluster analysis in R: Unsupervised machine learning (Vol. 1). Sthda.
5. *Wang, L.* (2016). Discovering phase transitions with unsupervised learning. Physical Review B, 94 (19), 195–105.
6. *Morris, G. M., & Lim-Wilby, M.* (2008). Molecular docking. In Molecular modeling of proteins (PP. 365–382). Humana Press.
7. *Ahmed, S., Ali, M. U., Ferzund, J., Sarwar, M. A., Rehman, A., & Mehmood, A.* (2017). Modern data formats for big bioinformatics data analytics. arXiv preprint arXiv:1707.05364.
8. *Van Rossum, G.* (2007, June). Python Programming Language. In USENIX annual technical conference (Vol. 41, P. 36).
9. *Korobkin, D. M., Fomenkov, S. A., Kolesnikov, S. G., &Orlova, Y. A.* (2013). A multi-stage algorithm for text documents filtering based on physical knowledge. World Applied Sciences Journal, 24 (24), 91–97.
10. *Wernand, M. R., Hommersom, A., & van der Woerd, H. J.* (2013). MERIS-based ocean colour classification with the discrete Forel–Ule scale. Ocean Science, 9 (3), 477–487.

11. *Raschka, S.* (2017). BioPandas: Working with molecular structures in pandas DataFrames. Journal of Open Source Software, 2(14), 279.
12. *McKinney, W.* (2011). pandas: a foundational Python library for data analysis and statistics. Python for High Performance and Scientific Computing, 14(9), 1-9.
13. *Oliphant, T.E.* (2006). A guide to NumPy (Vol. 1. P. 85). USA: Trelgol Publishing.
14. *Ari, N., &Ustazhanov, M.* (2014, September). Matplotlib in python. In 2014 11th International Conference on Electronics, Computer and Computation (ICECCO) (PP. 1–6). IEEE.
15. *Rose, P. W., Beran, B., Bi, C., Bluhm, at all* (2010). The RCSB Protein Data Bank: redesigned web site and web services. Nucleic acids research, 39(suppl_1), D392-D401.
16. *Hunanyan L., Makichyan A., Ghamaryan V., Hrutyunyan A., Danagulyan G.* (2019). Preliminary evaluation of the biological activityof benzo[4′,5′] imidazo[2′,1′:6,1]pyrido[2,3-d]pyrimidinederivatives by the methods of molecular modeling, Chemical Journal of Armenia, 72, № 4. PP. 530–533
17. *Trott, O., & Olson, A.J.* (2010). AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. Journal of computational chemistry, 31(2), 455–461.
18. *Li, H., Leung, K.S., Wong, M.H., & Ballester, P.J.* (2015). Improving AutoDock Vina using random forest: the growing accuracy of binding affinity prediction by the effective exploitation of larger data sets. Molecular informatics, 34 (2–3), 115–126.
19. *Biovia, D.S.* (2017). Discovery studio visualizer. San Diego, CA, USA, 936.
20. *Watkins, A. M., Rangan, R., & Das, R.* (2020). FARFAR2: Improved de novo Rosetta prediction of complex global RNA folds. Structure, 28(8), 963–976.

PROGRAM DEVELOPMENT FOR THE CONFORMATION ANALYSIS AND VISUALIZATION OF THE RESULTS OF MOLECULAR DOCKING USING CLUSTERING METHODS

V. Ghamaryan, Ch. Movsisyan, L. Hunanyan, A. Hovakimyan

Russian-Armenian University

ABSTRACT

One of the main tasks of modern bioinformatics is the analysis of big data obtained from medical-biological studies. Within the framework of the work, a system has been created that automatically classifies and visualizes the data obtained as a result of molecular docking with the help of clustering methods, ensuring their accurate selection and processing.

Keywords Bioinformatics, molecular docking, Auto Dock vina, FOREL, clustering analysis.

СВЕДЕНИЯ ОБ АВТОРАХ

Аветисян К.И.	магистрант второго курса по направлению подготовки «Прикладная математика и информатика» РАУ
Акопян А.Р.	магистрант второго курса по направлению подготовки «Прикладная математика и информатика» РАУ
Акопян Дж.А.	преподаватель кафедры системного программирования РАУ
Арамян Р.Г.	д.ф.-м.н., профессор, заведующий кафедрой математической кибернетики РАУ
Арутюнян М.С.	аспирант РАУ по специальности «Математическое и программное обеспечение вычислительных машин, комплексов и компьютерных сетей»
Бабаян А.В.	магистрант второго курса по специальности «Компьютерная инженерия» НПУА
Гарибян А.Г.	аспирант третьего года обучения кафедры дискретной математики и теоретической информатики ЕГУ
Гишян К.	магистрант Департамента компьютерных наук Батского университета, Великобритания
Григорян А.Т.	магистрант второго курса Института информационных и телекоммуникационных технологий НПУА
Григорян Г.Г.	к.э.н., Шанхайский университет финансов и экономики
Гукасян Ц.Т.	аспирант кафедры системного программирования РАУ, заведующий лабораторией им. В.П. Иванникова
Дарбинян А.А.	к.ф.-м.н., заведующий кафедрой математики и математического моделирования РАУ

Зограбян О.З.	студент четвертого курса факультета математики и механики ЕГУ
Казарян А.	аспирант ЕГУ
Казарян Д.А.	студент четвертого курса Института информационных и телекоммуникационных технологий НПУА
Камарян В.С.	студент пятого курса по специальности «Биоинженерия и биоинформатика» РАУ
Кроян А.К.	преподаватель кафедры высшей математики и физики Национального аграрного университета Армении
Мнацаканян В.А.	аспирант кафедры математической кибернетики РАУ
Мовсисян Ч.Г.	магистрант ЕГУ, кафедра программирования и информационных технологий
Овакимян А.С.	к.т.н., доцент
Оганнисян Р.А.	магистрант второго курса по специальности «Дискретная математики и теоретическая ин-форматика» ЕГУ
Смбатян Х.С.	магистрант второго курса по специальности «Дискретная математика и теоретическая информатика» ЕГУ
Степанян А.Г.	магистрант первого курса Института информационных и телекоммуникационных технологий НПУА
Унанян Л.С.	к.б.н., ст.преподаватель кафедры медицинской биохимии и биотехнологии РАУ
Хачатрян Л.Х.	к.т.н., Институт информационных и телекоммуникационных технологий, НПУА
Хачатурян М.А.	магистрант второго курса по направлению подготовки «Прикладная математика и информатика» РАУ

СОДЕРЖАНИЕ

Главный редактор РНИ *– М.Э. Авакян*
Корректор *– Н.И. Маргарян*
Компьютерная верстка *– А.Г. Антонян*

Адрес Редакции научных изданий
Российско-Армянского
университета:

0051, г. Ереван, ул. Овсепа Эмина, 123
тел./факс: (+374 10) 27-70-52 (внутр. 42-02)
e-mail: redaction.rau@gmail.com

Заказ № 8
Подписано к печати 02.05.2021г.
Формат 70x100$^{1}/_{16}$. Бумага офсетная № 1.
Объем 11.25 усл. п.л. Тираж 100 экз.